

---

# MANUEL UTILISATEUR ET GUIDE D'INSTALLATION DE L'EXOCET

Pixel Sur Mer



Revision 1.0

## Table des matières

<b>1</b>	<b>Sécurité</b>	<b>7</b>
<b>2</b>	<b>Introduction</b>	<b>8</b>
<b>3</b>	<b>Cas d'utilisation de l'EXOCET</b>	<b>10</b>
3.1	Cas d'utilisation communs . . . . .	10
3.2	Exocet Blue . . . . .	12
3.3	Exocet Silver . . . . .	13
3.4	Exocet Gold . . . . .	13
3.5	Exocet Red . . . . .	14
<b>4</b>	<b>Spécifications matérielles</b>	<b>15</b>
4.1	Électrique . . . . .	15
4.2	Mécanique et Environnement . . . . .	15
4.3	Entrées/sorties . . . . .	15
4.4	Connecteurs . . . . .	16
4.5	LEDS en façade . . . . .	17
<b>5</b>	<b>Démarrage rapide</b>	<b>18</b>
5.1	Allumer l'Exocet . . . . .	18
5.2	Vérifier l'intégrité du produit . . . . .	18
5.3	Configuration du réseau de votre PC pour la connexion initiale . . . . .	18
5.4	Changer l'adresse IP de l'Exocet . . . . .	21
5.5	Synchroniser l'heure . . . . .	22
5.6	Aperçu de l'interface utilisateur . . . . .	24
<b>6</b>	<b>Manta</b>	<b>25</b>
6.1	Aperçu de Manta . . . . .	25
6.2	Graphique <i>Hello world</i> de Manta . . . . .	27
6.3	Code couleur des boîtes Manta . . . . .	27
6.4	Flux manta . . . . .	28
6.5	boîte Manta . . . . .	29
6.6	boîte de configuration Manta . . . . .	33
6.7	Le déploiement . . . . .	34
6.8	Les messages . . . . .	35
6.9	Les manipulations de messages . . . . .	36
6.10	Traitement Manta . . . . .	41

6.11	Enregistrement de données . . . . .	42
6.12	Scripting Python . . . . .	42
6.13	Bibliothèques Manta . . . . .	45
6.14	Les bonnes pratiques de Manta . . . . .	46
<b>7</b>	<b>Application Web Exocet</b>	<b>52</b>
7.1	Aperçu de l'application Web Exocet . . . . .	52
7.2	Page « Statut » de l'application Web Exocet . . . . .	53
7.3	Page « Enregistrements » de l'application Web Exocet . . . . .	55
7.4	Page « Table de données » de l'application Web Exocet . . . . .	56
7.5	Page « Alarmes » de l'application Web Exocet . . . . .	57
7.6	Tableaux de bord de l'application Web Exocet . . . . .	59
<b>8</b>	<b>Communication avec des logiciels tiers</b>	<b>63</b>
<b>9</b>	<b>Tutoriel : du capteur au cloud</b>	<b>64</b>
9.1	Lire les données analogiques . . . . .	64
9.2	Visualiser les données dans une probe . . . . .	66
9.3	Renommer les données d'entrée . . . . .	66
9.4	Calibrer l'entrée . . . . .	68
9.5	Visualiser les données dans le tableau de données . . . . .	69
9.6	Visualiser les données dans un tableau de bord . . . . .	70
9.7	Configuration d'une alerte . . . . .	73
9.8	Visualiser l'alerte . . . . .	74
9.9	Enregistrement des données . . . . .	75
9.10	Envoyer des données au cloud . . . . .	76
9.11	Visualiser les données cloud avec Grafana . . . . .	79
<b>10</b>	<b>Manta boxes overview table</b>	<b>80</b>
10.1	Input interfaces . . . . .	80
10.2	Input devices . . . . .	81
10.3	Input softwares . . . . .	82
10.4	Input . . . . .	82
10.5	Input User . . . . .	82
10.6	Protocol Decode . . . . .	83
10.7	Processing . . . . .	84
10.8	Output Interface . . . . .	86
10.9	Protocole Encode . . . . .	87
10.10	Output Device . . . . .	87

10.11 Output Software . . . . .	88
10.12 Output . . . . .	88
10.13 Debug . . . . .	88
10.14 wind . . . . .	89
<b>11 Manta boxes detailed description</b>	<b>90</b>
11.1 add_delay . . . . .	90
11.2 adrena_xdr . . . . .	91
11.3 ahrs . . . . .	91
11.4 einstein . . . . .	93
11.5 ais . . . . .	94
11.6 alarm . . . . .	94
11.7 analog . . . . .	95
11.8 antiCapsize . . . . .	96
11.9 ascii_out . . . . .	99
11.10 blinkManager . . . . .	100
11.11 bravo_in . . . . .	101
11.12 bravo_out . . . . .	102
11.13 bretronics . . . . .	102
11.14 button . . . . .	104
11.15 calibration . . . . .	105
11.16 canbus_in . . . . .	108
11.17 canbus_out . . . . .	109
11.18 cycle . . . . .	110
11.19 damping . . . . .	111
11.20 datalogger . . . . .	114
11.21 debounce . . . . .	115
11.22 display_bandg . . . . .	116
11.23 display_garmin . . . . .	117
11.24 display_nke . . . . .	122
11.25 e_telltales . . . . .	124
11.26 empirbus . . . . .	124
11.27 empirbus_out . . . . .	126
11.28 enlight . . . . .	127
11.29 eta . . . . .	128
11.30 exocet_in . . . . .	129
11.31 exocet_out . . . . .	130
11.32 expression . . . . .	131



11.33 fail_safe . . . . .	139
11.34 freq_divider . . . . .	143
11.35 gpio_in . . . . .	144
11.36 gpio_out . . . . .	145
11.37 h5000 . . . . .	146
11.38 height_estimation . . . . .	147
11.39 hyperion . . . . .	152
11.40 idatanet . . . . .	153
11.41 ixblue . . . . .	153
11.42 json_dec . . . . .	159
11.43 json_enc . . . . .	159
11.44 keel . . . . .	160
11.45 kvh . . . . .	162
11.46 lee_cur_tools . . . . .	164
11.47 manta_status . . . . .	170
11.48 modbus_in . . . . .	171
11.49 modbus_out . . . . .	172
11.50 moving_mean . . . . .	174
11.51 net2000_in . . . . .	176
11.52 net2000_out . . . . .	176
11.53 nmea0183_dec . . . . .	176
11.54 nortek . . . . .	177
11.55 obstacleManager . . . . .	178
11.56 on_board_assistant . . . . .	180
11.57 opc_in . . . . .	181
11.58 oscar . . . . .	182
11.59 overlay_manager . . . . .	183
11.60 pid . . . . .	185
11.61 pilot_overlay_bandg . . . . .	192
11.62 pilot_overlay_nke . . . . .	195
11.63 pilot_settings_bandg . . . . .	198
11.64 polar . . . . .	199
11.65 probe . . . . .	203
11.66 python_function . . . . .	203
11.67 regulation_performance . . . . .	212
11.68 ride_height . . . . .	213
11.69 sailnet . . . . .	215
11.70 sbg . . . . .	216

11.71 scalar . . . . .	217
11.72 senix . . . . .	218
11.73 sentea . . . . .	219
11.74 serial . . . . .	220
11.75 serial_in . . . . .	221
11.76 sirius . . . . .	222
11.77 stats . . . . .	222
11.78 string . . . . .	224
11.79 system . . . . .	225
11.80 table . . . . .	227
11.81 tcp_in . . . . .	228
11.82 tcp_out . . . . .	229
11.83 tick . . . . .	229
11.84 udp_in . . . . .	230
11.85 udp_out . . . . .	231
11.86 varfilter . . . . .	231
11.87 varmux . . . . .	232
11.88 vector_man . . . . .	233
11.89 victron . . . . .	235
11.90 wattandsea . . . . .	236
11.91 williamson . . . . .	239
11.92 wind_apparent . . . . .	240
11.93 wind_ground . . . . .	243
11.94 wind_mastrot . . . . .	245
11.95 wind_motion . . . . .	247
11.96 wind_processor_hr . . . . .	250
11.97 wind_true . . . . .	251
11.98 wind_upwash . . . . .	254
11.99 wtp . . . . .	258
11.100wtpSerialManager . . . . .	259
11.101zeromq_in . . . . .	260
11.102zeromq_out . . . . .	261
<b>12 Appendix</b>	<b>262</b>
12.1 Récupérer l'adresse IP de l'Exocet . . . . .	262
12.2 Configuration de votre PC en tant que serveur de temps . . . . .	262

## 1 Sécurité

L'appareil est conforme à la section 15 des règles de la FCC. Son fonctionnement est soumis aux deux conditions suivantes : (1) Cet appareil ne doit pas causer d'interférences nuisibles, et (2) Cet appareil doit accepter toute interférence reçue, y compris les interférences qui causent un fonctionnement indésirable.

La société « Pixel Sur Mer » n'est pas responsable des changements ou modifications non expressément approuvés par la partie responsable de la conformité. De telles modifications pourraient annuler le droit de l'utilisateur à faire fonctionner l'équipement.

NOTE : Cet équipement a été testé et déclaré conforme aux limites d'un appareil numérique de classe A, conformément à la section 15 des règles de la FCC. Ces limites sont conçues pour fournir une protection raisonnable contre les interférences nuisibles lorsque l'équipement est utilisé dans un environnement commercial. Cet équipement génère, utilise et peut émettre de l'énergie radiofréquence et, s'il n'est pas installé et utilisé conformément au manuel d'instructions, peut causer des interférences nuisibles aux communications radio. L'utilisation de cet équipement dans une zone résidentielle est susceptible de provoquer des interférences nuisibles, auquel cas l'utilisateur devra corriger ces interférences à ses propres frais.

## 2 Introduction

Les skippers d'aujourd'hui et leurs ingénieurs concepteurs sont confrontés au grand défi technologique des bateaux volants et des navires à haute performance. Afin de compléter le panel d'équipements existants et de répondre pleinement aux nouvelles demandes de ses clients, PIXEL SUR MER a innové et développé des systèmes parmi les plus avancés du marché : **La ligne EXOCET**.

La ligne Exocet est composée de 4 produits différents qui ont chacun leurs propres fonctions :

- **Exocet Blue** : Acquisition et surveillance de données universelles
- **Exocet Silver** : Processeur central pour améliorer la sécurité et les performances des bateaux de course de dernière génération :
  - Gestion avancée des capteurs avec support « failsafe »
  - Calcul avancé du vent réel
  - Amélioration de la sécurité et des performances du pilote automatique avec des règles de surcouches de pilotage
- **Exocet Gold** : Unité de calcul pour contrôle de vol
- **Exocet Red** : Servocommande de puissance de haute précision contrôlé par l'**Exocet Gold**.



**FIG. 1** : Ligne Exocet

Les produits Exocet sont compatibles et s'interfaçent avec les systèmes de navigation les plus performants sur n'importe quel bateau. Mais, ils ne sont pas limités à être utilisés dans un environnement système de bateau. En particulier, l'Exocet Blue est un enregistreur de données universel qui répond aux exigences et aux besoins **industriels** et **aéronautiques**.



**L'Exocet Red** est un produit spécifique non décrit dans ce document. Veuillez vous reporter au manuel utilisateur de l'Exocet Red pour plus d'informations.

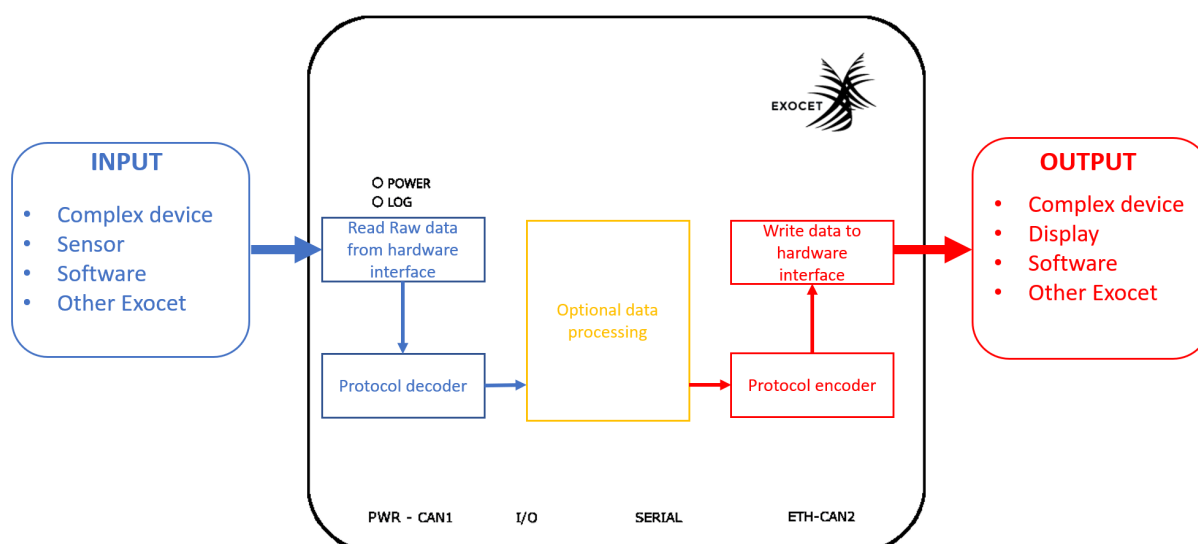
### 3 Cas d'utilisation de l'EXOCET

Les produits Exocet peuvent être utilisés pour un ou plusieurs cas d'utilisation en même temps. Certains sont applicable à tous les produits, d'autres sont spécifiques aux modèles Blue, Silver ou Gold.

#### 3.1 Cas d'utilisation communs

##### Passerelle de données

Les produits Exocet peuvent lire des données à partir de diverses interfaces matérielles et protocoles, puis les transmettre à d'autres équipements ou logiciels utilisant un autre protocole et une autre interface matérielle.



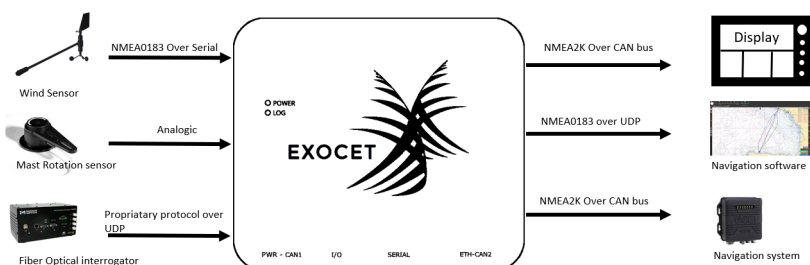
**FIG. 2 :** Concept de la passerelle de données

- **Interfaces matérielles d'entrée :** Analogique, GPIO, série, Ethernet/UDP/TCP, bus CAN
- **Décodeur de protocole d'entrée standard :** NMEA0183, NMEA2K, J1939, Modbus
- **Décodeur de protocole d'entrée propriétaire :**
  - **Systèmes de navigation :** B&G H5000 CPU et WTP3, NKE ProcessorHR, Bravo 4 CPU
  - **Statut du pilote automatique :** Pilote B&G H5000, NKE Gyropilot2
  - **Surveillance de l'alimentation électrique :** Générateur aéro et hydro Watt&Sea, contrôleur de panneau solaire Victron, batterie Williamson et Bren-Tronics, ETA Powerplex, Garmin EmpirBus.
  - **Fibre optique :** Interrogateur de fibre optique Luna et Pixel Sur Mer, logiciel Luna Enlight.

- **Capteur d'attitude** : lxBlue, SBG, KVH
- **Exocet** : recevoir une donnée d'un autre produit Exocet
- **Divers** : Nortek DVL, E-telltales, Sirius (bus de capteurs GPIO Pixel sur Mer), Oscar, clavier Blink...
- **Traitement des données** : amortissement, calibration, moyenne, variance, min/max, médiane, retard, sous-échantillonnage...
- **Interfaces matérielles de sortie** : GPIO, série, Ethernet/UDP/TCP, bus CAN
- **Encodeur de protocole de sortie standard** : NMEA0183, NMEA2K, J1939, Modbus
- **Encodeur de protocole de sortie propriétaire** :
  - **Système de navigation** : NKE ProcessorHR, Bravo 4 CPU
  - **Contrôle du pilote automatique** : B&G H5000 pilot, NKE Gyropilot2
  - **Écran** : B&G, NKE, Garmin...
  - **Exocet** : envoyer des données vers un autre produit Exocet
  - **Divers** : Adrena, Expedition, Idatanet, Garmin EmpirBus...



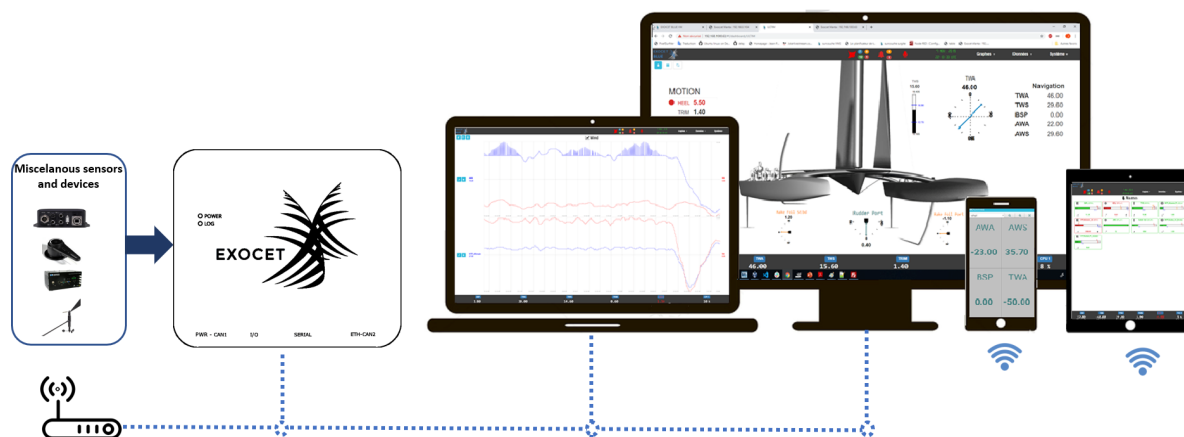
Avec le **support de scripts Python**, les possibilités des Exocet peuvent facilement être étendues avec tout décodeur/encodeur de protocole personnalisé ou toute fonction de traitement de données personnalisée.



**FIG. 3 :** Exemple de passerelle de données

## Surveillance de système

Des tableaux de bord et des graphiques personnalisés peuvent être créés pour surveiller les données en temps réel. Un éditeur est disponible pour réaliser des interfaces graphiques adaptées à chaque besoin. Des seuils d'alarme peuvent être définis afin d'être notifié lorsqu'une donnée est en dehors de sa plage de valeurs attendues. L'interface utilisateur est basée sur le web, elle est donc accessible depuis un PC, une tablette ou un smartphone. Si la connexion est établie depuis un smartphone, une application web adaptée à la petite taille de l'écran est lancée.



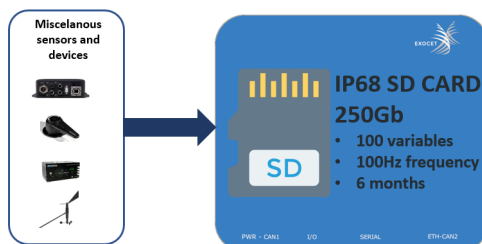
**FIG. 4 :** Suivi des données en direct

### 3.2 Exocet Blue

Première innovation de PIXEL SUR MER, **Exocet Blue** est un système embarqué d'acquisition, de stockage et de supervision de données à la pointe de la technologie. Avec sa capacité de stockage de 250Gb, il peut enregistrer jusqu'à 100 variables, avec une fréquence d'acquisition de 100Hz pendant au moins 6 mois.

#### boite noire

L'Exocet Blue peut être utilisé comme une **boite noire** afin de faciliter l'investigation des incidents. Comme il commence à enregistrer automatiquement lorsqu'il est allumé, il ne nécessite aucune interaction avec l'utilisateur. La carte SD de stockage des données est classée **IP68** pour la protection contre l'immersion dans l'eau.

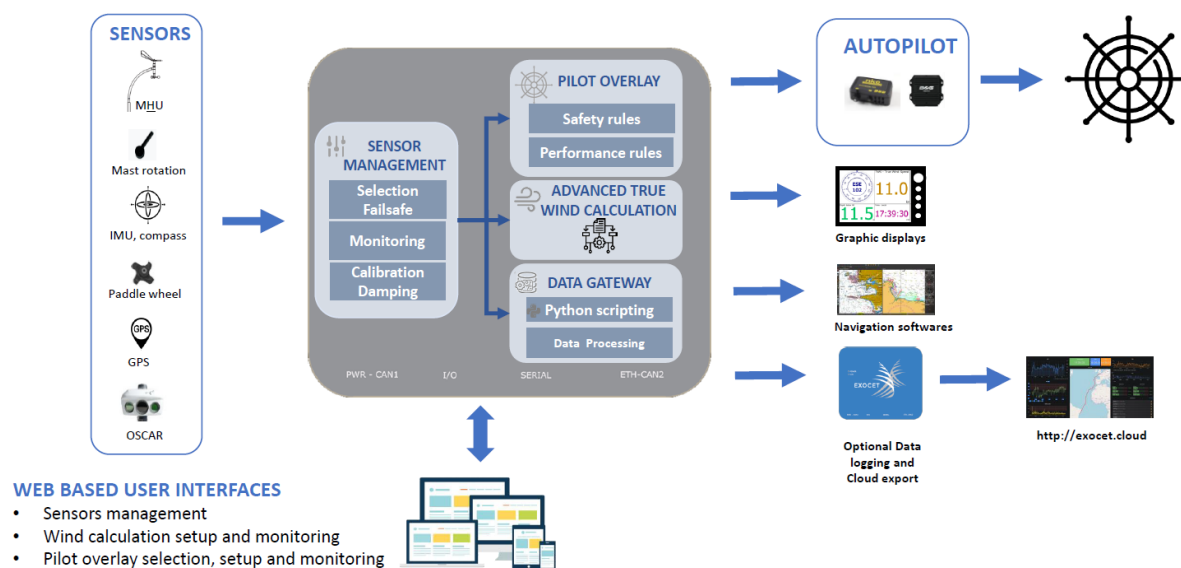


**FIG. 5 :** boite noire



### 3.3 Exocet Silver

Système de navigation autonome pour améliorer la sécurité et les performances des bateaux de course de dernière génération.



**FIG. 6 :** Aperçu de l'Exocet Silver

- Gestion avancée des capteurs avec support « failsafe »
- Calcul avancé du vent réel
- Amélioration de la sécurité et des performances du pilote automatique avec des règles de surcouches de pilotage
- Compatibilité avec la plupart des protocoles, équipements et écrans marins
- Interfaces graphiques conviviales

### 3.4 Exocet Gold

**EXOCET GOLD signifie :**

- Unité de calcul pour :
  - Lire et décoder des capteurs haute fréquence
  - Gérer les algorithmes de contrôle de vol
  - Commander un ou plusieurs appendices via l'EXOCET RED

### **3.5 Exocet Red**

Servocommande rapide et précis pour le contrôle local d'appendices via un moteur de pompe hydraulique ou des vérins électriques. Mode de sécurité intégré.

## 4 Spécifications matérielles

### 4.1 Électrique

- **Tension d'entrée** : 9-51V
- **Consommation d'énergie** : 4W

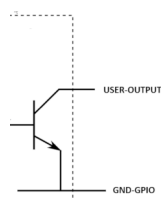
Les produits Exocet utilisent des condensateurs pour se prémunir des sauts de tension.

### 4.2 Mécanique et Environnement

- **Poids** : 350 g
- **Matériau** : Aluminium anodisé
- **Taille** : 113 x 97 x 40mm
- **Température spécifiée** : -10/+55°
- **Classement IP** : IP67
- **Stockage persistant** : Carte SD IP68 (Exocet Blue uniquement)

### 4.3 Entrées/sorties

Interface	Numéro	Détails
Ethernet	1	Gbe (Giga byte ethernet)
CAN 2.0	2	Jusqu'à 1 mbps
Entrées analogiques	6	Isolées - 16 bits (0/5V, +/-5V, 0/10V, +/-10V). Echantillonnage jusqu'à 50Hz
Entrées numériques	2	Isolées
Sorties numériques	3	Isolées, collecteur ouvert. 1A max.
Série	6	5 RS232, 1 RS232/RS422. Jusqu'à 115200 Bds
Alimentation	2	Accès à la tension d'entrée pour faciliter le câblage



**FIG. 7 :** GPIO collecteur ouvert

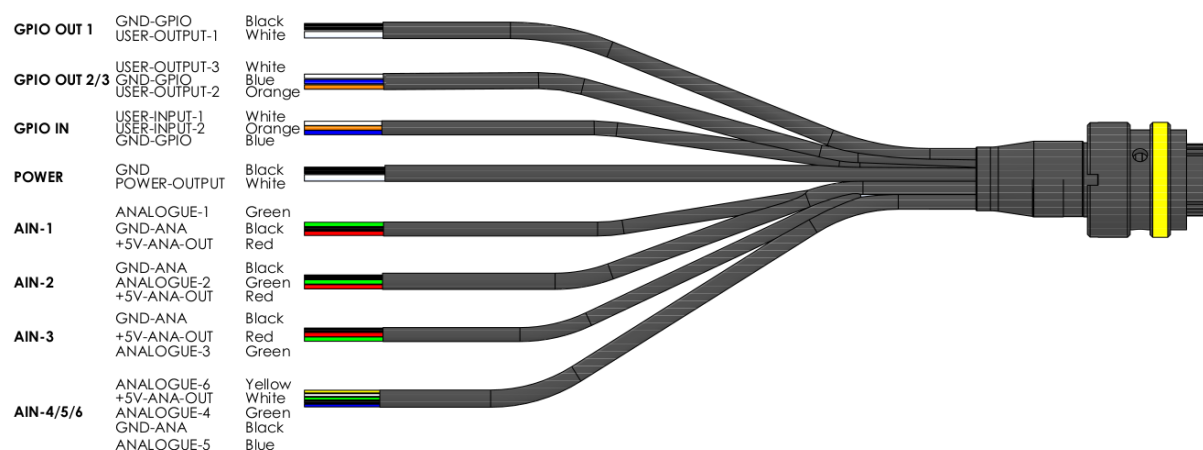
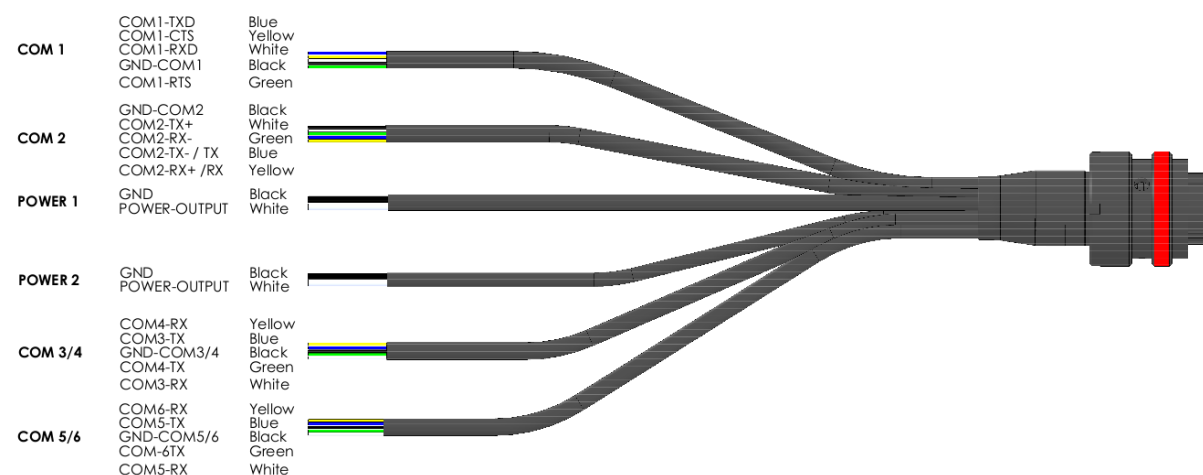
## 4.4 Connecteurs



**FIG. 8 :** Interface matérielle

Les Exocet Blue, Silver et Gold ont 4 connecteurs :

- Un connecteur mâle standard NMEA2000 pour l'alimentation et CAN1
- Un connecteur jaune I/O pour les entrées/sorties numérique et analogique (voir détails ci-dessous)
- Un connecteur rouge Serial pour les entrées/sorties série (voir détails ci-dessous).
- Un petit connecteur rouge relié à des connecteurs standards Ethernet et NMEA2000 (fourni)

**FIG. 9 :** Câble E/S**FIG. 10 :** Câble série

## 4.5 LEDS en façade

Chaque Exocet possède deux voyants, un vert et un rouge.

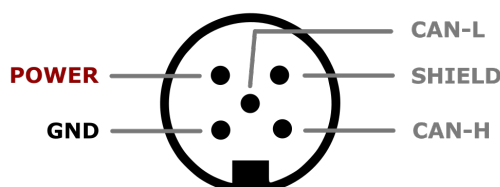
La led verte allumée indique que l'Exocet est en fonctionnement.

La led rouge allumée indique que le datalogger est en train d'enregistrer. La led rouge clignotante indique qu'une boîte Manta est en statut erreur.

## 5 Démarrage rapide

### 5.1 Allumer l'Exocet

La première étape consiste à allumer l'Exocet. Comme mentionné dans les spécifications électriques, l'alimentation est comprise entre 9 et 51 volts via le connecteur CAN-1. Il s'agit d'un connecteur mâle CAN M12 standard. Voici un schéma du brochage :



**FIG. 11 :** Brochage CAN-1

Le câble d'alimentation fourni avec l'Exocet peut être utilisé pour connecter l'Exocet à un bus Can alimenté.

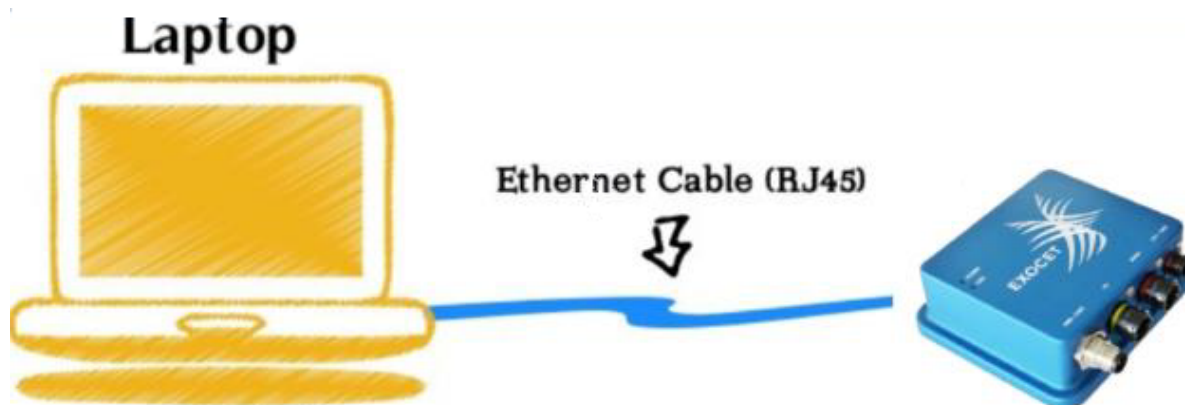
### 5.2 Vérifier l'intégrité du produit

Du côté matériel, vérifiez que le sceau adhésif est présent sur une vis à l'arrière de l'Exocet. Du côté logiciel, vérifiez que le voyant vert étiqueté POWER est allumé.

### 5.3 Configuration du réseau de votre PC pour la connexion initiale

Les Exocet sont fournis avec une adresse IP **192.168.100.251**.

Pour éviter tout problème de réseau, il est recommandé de connecter directement l'Exocet à un PC. Pour pouvoir communiquer avec l'Exocet, l'adresse IP du PC doit être réglée sur 192.168.100.XXX avec XXX un nombre de 1 à 250.



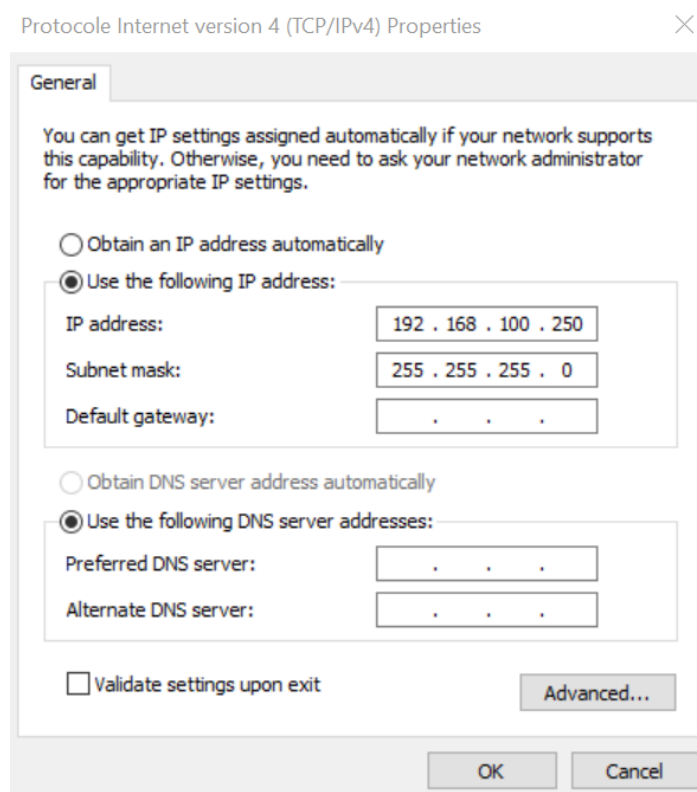
**FIG. 12 :** connexion directe de l'ordinateur portable

### **Configurez votre PC avec l'adresse IP statique 192.168.100.250**

Pour définir une adresse IP statique dans Windows 7, 8 et 10 :

1. Cliquez sur **Menu Démarrer > Panneau de configuration > Centre Réseau et Partage ou Réseau et Internet > Centre Réseau et Partage.**
2. Cliquez sur **Modifier les paramètres de l'adaptateur.**
3. Cliquez avec le bouton droit de la souris sur **Connexion au réseau local.**
4. Cliquez sur **Propriétés.**
5. Sélectionnez **Protocole Internet version 4 (TCP/IPv4).**
6. Cliquez sur **Propriétés.**
7. Sélectionnez **Utiliser l'adresse IP suivante.**
8. Saisissez l'adresse IP **192.168.100.250**, masque de sous-réseau **255.255.255.0.**
9. Cliquez sur **OK.**

Votre ordinateur utilise maintenant l'adresse IP statique 192.168.100.250.

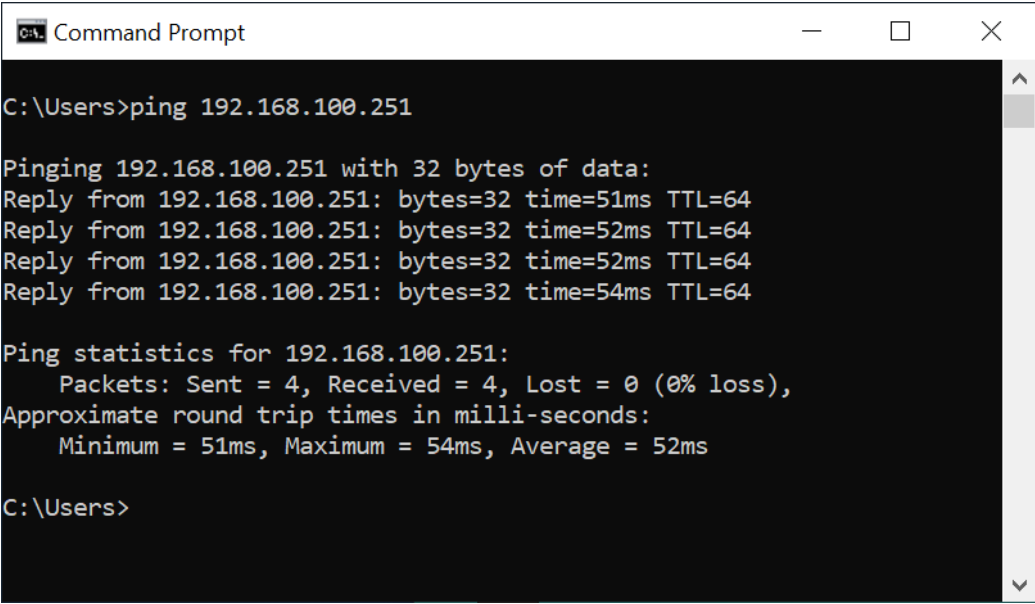


**FIG. 13 :** Réglage IPV4

### Vérifier la communication avec EXOCET

1. Cliquez sur **Menu Démarrer > Invite de commande**.
2. Dans la fenêtre d'invite de commande, entrez la commande **ping 192.168.100.251** et appuyez sur la touche **ENTER**.





```
Command Prompt

C:\Users>ping 192.168.100.251

Pinging 192.168.100.251 with 32 bytes of data:
Reply from 192.168.100.251: bytes=32 time=51ms TTL=64
Reply from 192.168.100.251: bytes=32 time=52ms TTL=64
Reply from 192.168.100.251: bytes=32 time=52ms TTL=64
Reply from 192.168.100.251: bytes=32 time=54ms TTL=64

Ping statistics for 192.168.100.251:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 51ms, Maximum = 54ms, Average = 52ms

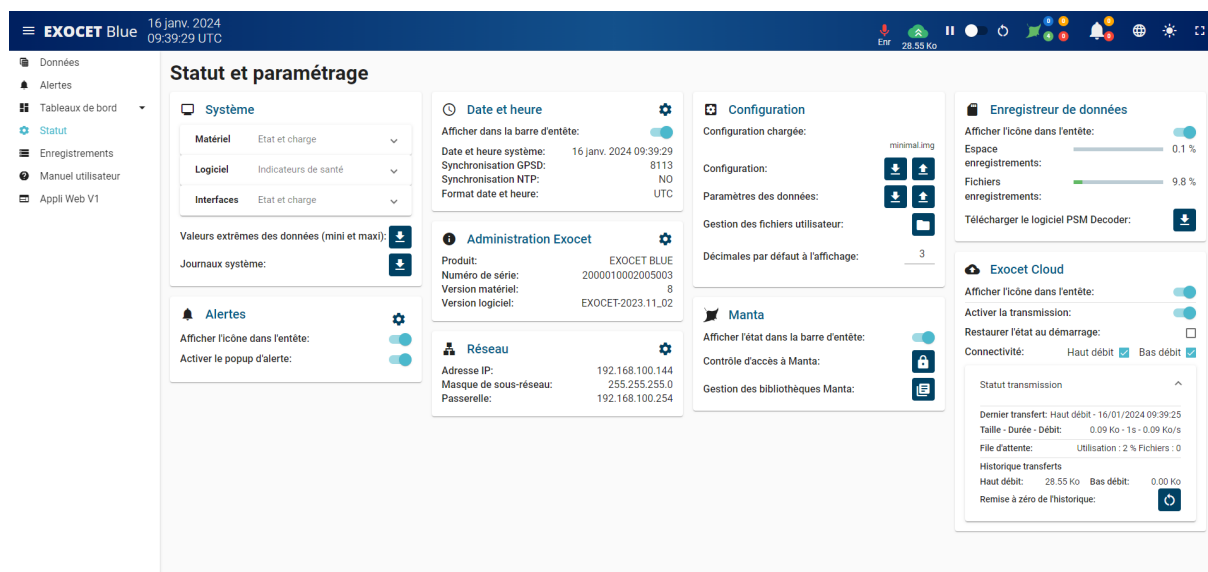
C:\Users>
```

**FIG. 14 :** commande ping

Si une réponse est reçue, le PC et l'Exocet sont connectés, sinon, vérifiez les connexions physiques.

## 5.4 Changer l'adresse IP de l'Exocet

Sur votre navigateur, recherchez 192.168.100.251 et vous devriez arriver sur l'interface web de l'Exocet, sur la page nommée « Statut ».



**Fig. 15 :** Page de réglages de l'application web Exocet

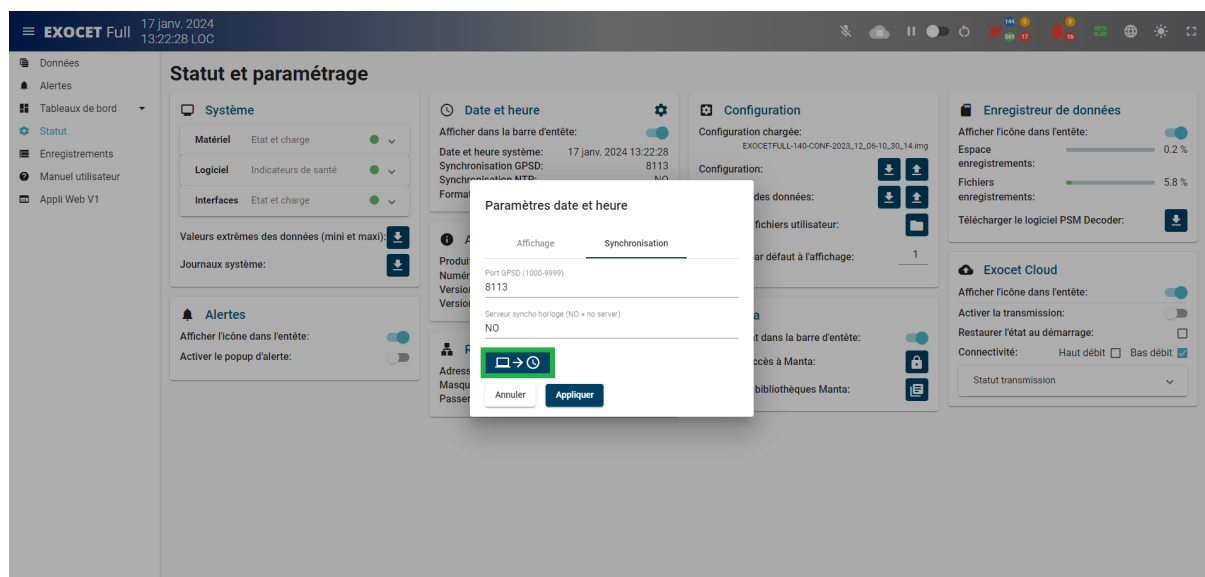
Dans la section réseau, cliquez sur la roue dentée pour accéder aux paramètres IP. Changez l'IP pour celle que vous désirez et cliquez sur « Ok » pour valider. L'Exocet redémarrera avec les nouveaux paramètres réseau.

## 5.5 Synchroniser l'heure

L'Exocet possède sa propre gestion du temps. La date et l'heure affichées en haut de l'application web peuvent différer de la date et de l'heure réelles. Il existe plusieurs possibilités pour changer l'heure d'un Exocet.

### Appliquer l'heure de l'ordinateur

Sur la même page web, allez dans les paramètres de date et d'heure en cliquant sur la roue dentée de la carte *Date et Heure*. Comme montré sur l'image ci-dessous, allez dans le menu *Synchronisation* et cliquez sur le bouton graphique pour appliquer l'heure du PC à l'Exocet (mis en évidence en vert) puis cliquez sur le bouton *Appliquer*.



**FIG. 16 :** Synchroniser l'heure avec l'ordinateur

L'heure affichée en haut à gauche de l'application web est maintenant la même que celle de votre ordinateur.

## Synchronisation GPS

Le temps à l'intérieur de l'Exocet peut dériver avec le temps. Il est recommandé d'utiliser une synchronisation dynamique pour le garder synchronisé. Dans le même menu que pour l'heure de l'ordinateur, vous pouvez choisir un *port GPSD* pour envoyer une phrase NMEA0183 \$RMC et *Appliquer* les changements. Ainsi, le temps de l'Exocet suivra constamment le temps GPS de haute précision. L'icône suivante apparaîtra à côté de la date et de l'heure sur l'application web :



**FIG. 17 :** Synchronisation GPS

## Serveur NTP

Une autre manière de rester à jour est de se synchroniser à un serveur NTP. Dans le même menu que pour l'heure de l'ordinateur, vous pouvez remplir une adresse IP de serveur NTP dans le champ *Serveur synchro horloge* et *Appliquer* les changements. Le temps de l'Exocet suivra constamment le temps du serveur. L'icône suivante apparaîtra à côté de la date et de l'heure sur l'application web :



**FIG. 18 :** Serveur NTP

Note : Chaque Exocet possède un serveur NTP interne. Une fois correctement synchronisé, un Exocet peut être utilisé comme référence temporelle pour d'autres appareils.

## 5.6 Aperçu de l'interface utilisateur

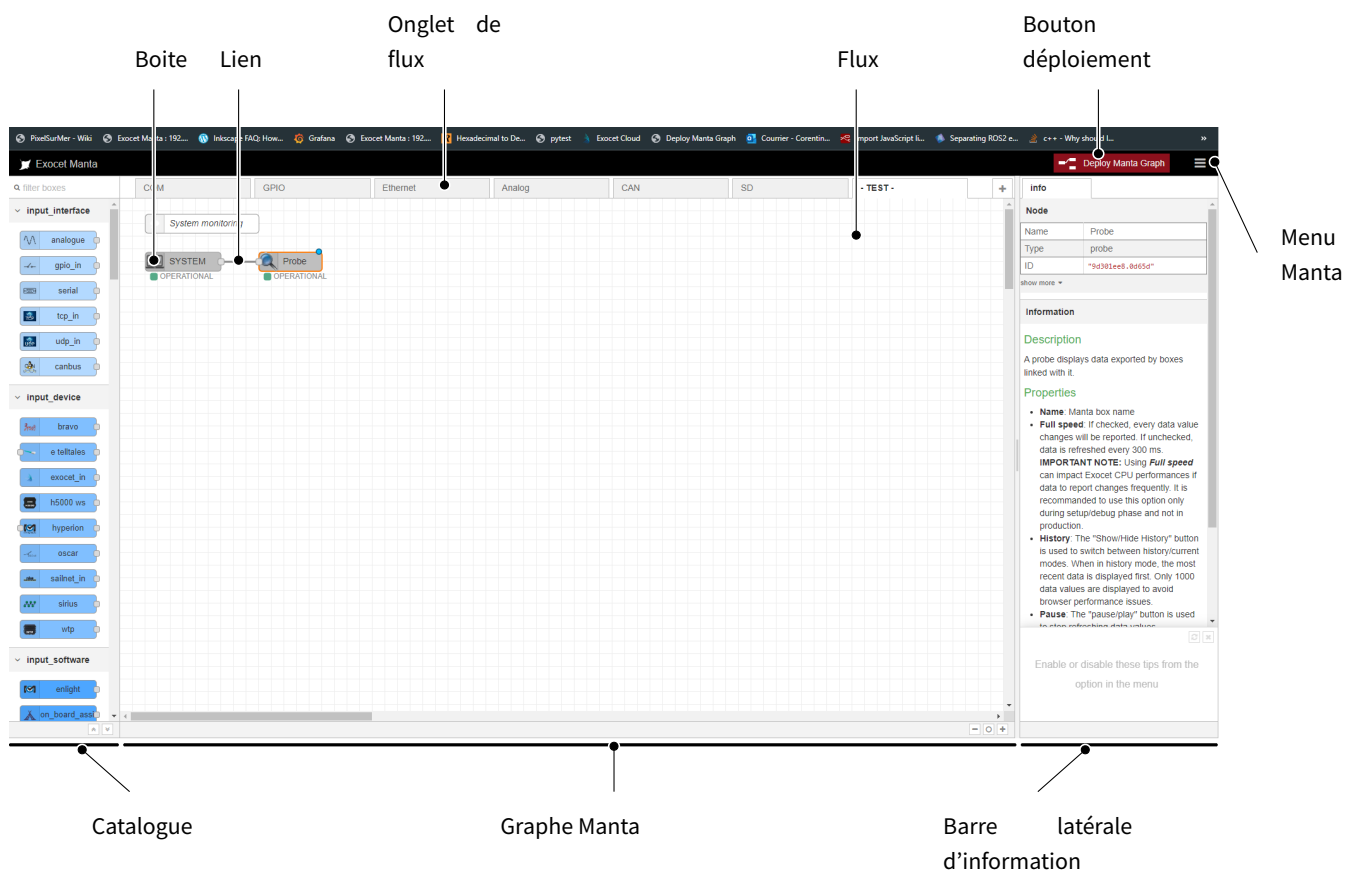
L'interface utilisateur d'Exocet est accessible en utilisant un navigateur web avec l'adresse IP de l'Exocet comme URL. Par défaut, l'adresse IP est 192.168.100.251. Cette interface utilisateur est composée de deux parties :

- *MANTA*, qui est notre outil de configuration graphique convivial :
  - Pour définir quelles données entrantes sont collectées et comment elles le sont.
  - Pour traiter les données collectées avec une grande variété de fonctions de traitement disponibles et par script python.
  - Définir quelles données sont exportées vers un équipement ou un logiciel tiers et comment elles le sont.
  - Pour sélectionner les données à enregistrer sur la carte SD ou à envoyer à Exocet Cloud (Exocet Blue uniquement).
- *Exocet Web App*, qui est l'interface utilisateur quotidienne :
  - Pour surveiller les données en direct grâce à des tableaux de bord, des graphiques et des affichages personnalisables. Un éditeur vous permet de composer les vues en direct les mieux adaptées à chaque besoin.
  - Définir et surveiller les alarmes avec plusieurs niveaux de criticité.
  - Gérer les fichiers d'enregistrement des données (Exocet Blue uniquement).
  - Configurer et surveiller les données internes de l'Exocet.

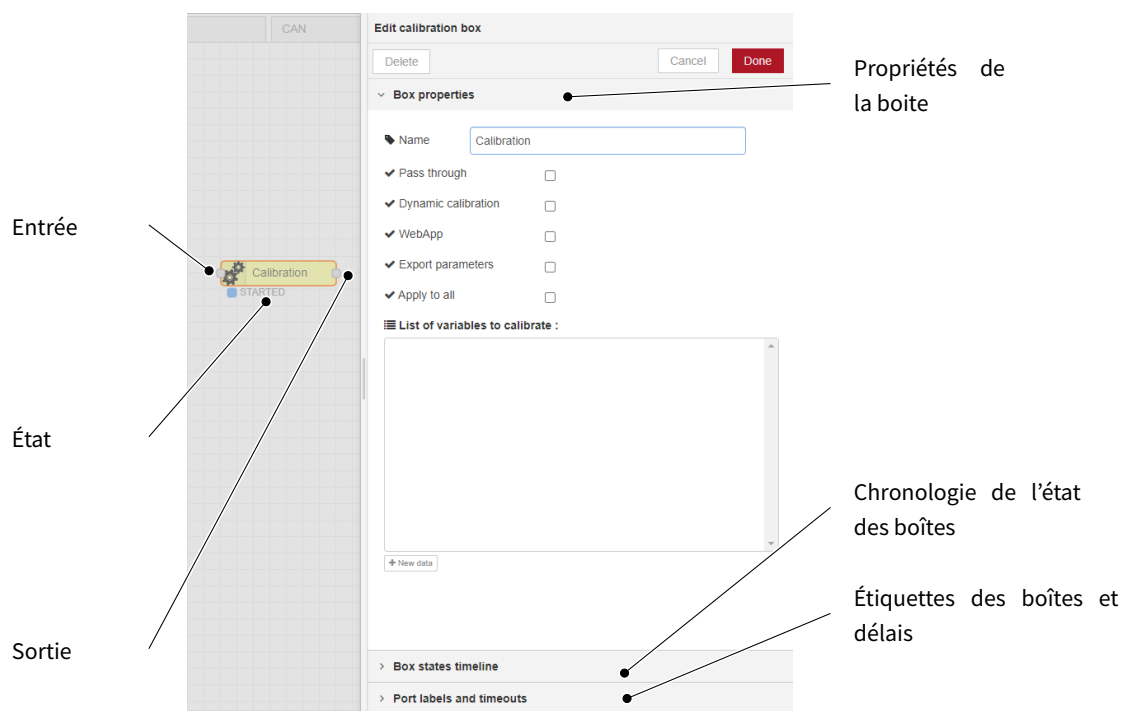
## 6 Manta

### 6.1 Aperçu de Manta

Manta est l'éditeur de configuration des produits de la gamme Exocet. Il se présente sous la forme d'un éditeur graphique de programmation de manipulation de données.



**FIG. 19 :** Aperçu de l'interface utilisateur de Manta.



**FIG. 20 :** Aperçu de la boîte Manta


Pour concevoir un graphique Manta, et ainsi configurer le traitement des données Exocet :

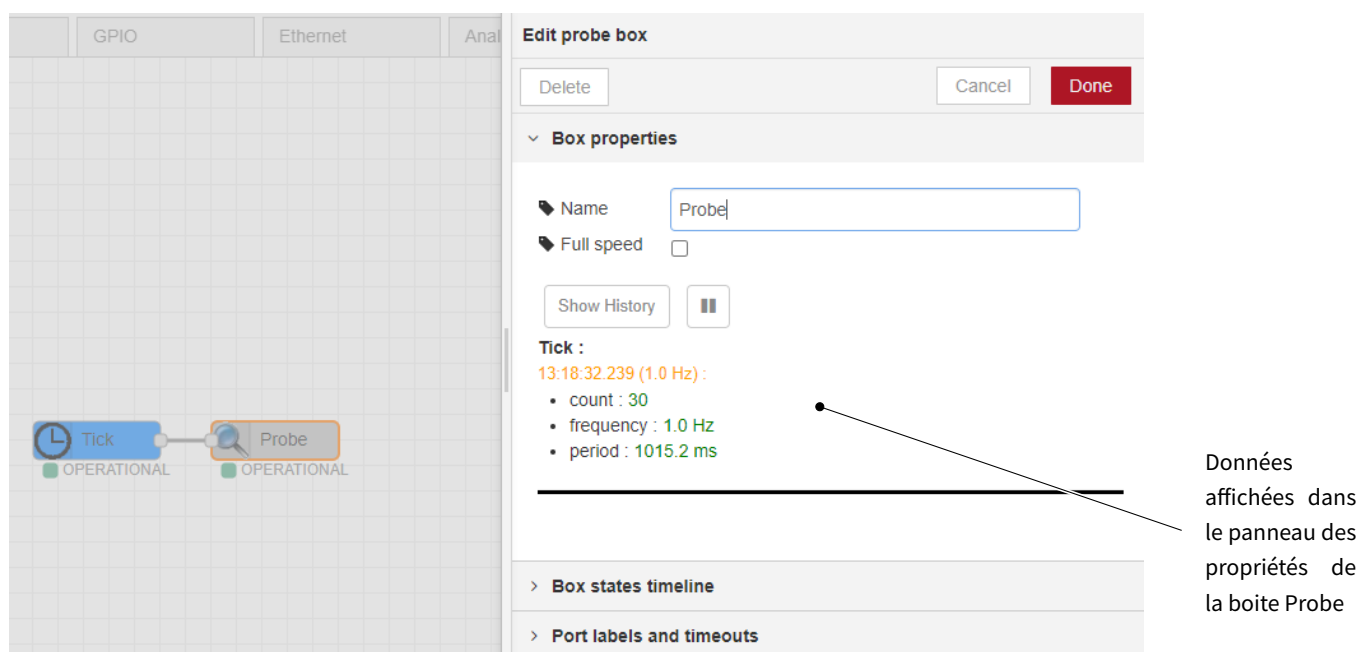
1. Faites glisser et déposez les *boîtes* du *catalogue* sur la page de flux. L'outil de recherche en haut du catalogue peut aider à rechercher une boîte spécifique.
2. Configurez chaque *boîte* en double cliquant pour ouvrir la *barre latérale d'édition*. Une information contextuelle est disponible dans la marge de droite.
3. Envoyez un « message » d'une boîte à une autre en reliant le *port de sortie* d'une boîte au *port d'entrée* d'une autre avec un *fil*. Pour ce faire, appuyez sur le bouton gauche de la souris sur le *port de sortie* d'une boîte, faites-le glisser vers le *port d'entrée* de la boîte de destination et relâchez le bouton de la souris.
4. Appuyez sur le bouton « Deploy Manta Graph » en haut à droite de l'espace de travail.



Note : Si une boîte n'est pas bien configurée et empêche le déploiement, un petit triangle orange apparaît dans le coin supérieur droit de la boîte. Toutes les boîtes modifiées depuis le dernier déploiement sont notifiées par un point bleu dans le coin supérieur droit des boîtes.

## 6.2 Graphique *Hello world* de Manta

1. Glissez et déposez une boîte *Tick* du *catalogue* vers le *flux*.
2. Glissez et déposez une boîte *Probe* du *catalogue* vers le *flux*.
3. Reliez les deux en appuyant sur le bouton gauche de la souris sur le *port de sortie* de la boîte *Tick*, en faisant glisser vers le *port d'entrée* de la boîte *Probe* et en relâchant le bouton de la souris.
4. Appuyez sur le bouton .
5. Double-cliquez sur la boîte *Probe*, les données de la boîte *Tick* devraient apparaître sur le *panneau de propriétés* de la boîte.



**Fig. 21 :** Affichage de données en direct à l'aide d'une boîte "Probe"

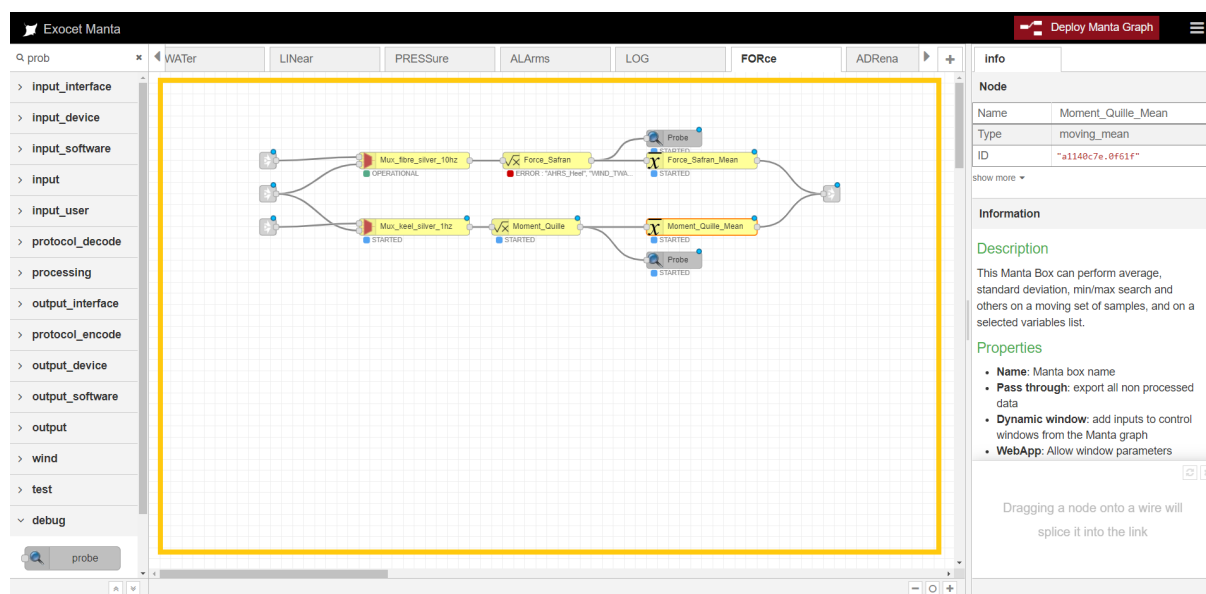
## 6.3 Code couleur des boîtes Manta

Dans le catalogue, les boîtes sont organisées avec le code couleur suivant :

- **Input :** Acquisition de données provenant de divers équipements, software ou entrées Exocet.
- **Protocol :** Décodage de protocols Ethernet, bus CAN, liaison série.
- **Processing :** Traitement des données. Calibration, calculs mathématiques, filtrage, programmation...
- **Output :** Export de données vers divers équipements, software ou sorties Exocet.
- **Debug :** Surveillance du comportement et visualisation

## 6.4 Flux manta

Un flux est une page contenant un ensemble de boites issues du catalogue et reliées entre elles par des fils.



**FIG. 22 :** Manta Flow

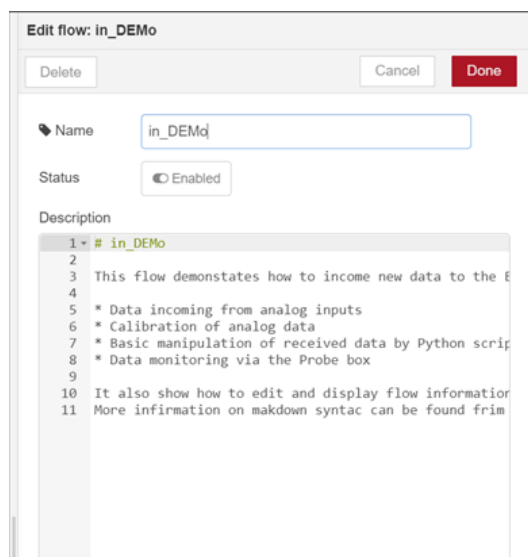
**Un flux est composé de :**

- Un onglet pour sélectionner le flux à afficher et voir son statut
- Une description visible dans le panneau latéral d'information
- Un graphique de boites et de fils
- Un statut : Activé ou Désactivé



**Paramètres :**

Accessible en double cliquant sur son onglet





**FIG. 23 :** Paramètres du flux Manta

- **Nom** : Nom unique du flux
- **Statut** : change l'état du flux. Les boîtes d'un flux à l'état désactivé ne sont pas exécutées. Les flux dans l'état désactivé sont identifiés par une icône  dans leur onglet.
- **Description** : éditer la description du flux au format markdown
- Le bouton  permet de supprimer le flux.

#### Ajouter un flux :

Cliquez sur le bouton  dans la barre d'onglets.

#### Réorganiser les flux :

Par glisser-déposer de l'onglet. La disposition des flux n'a pas d'impact sur l'ordre d'exécution des boîtes contenues.



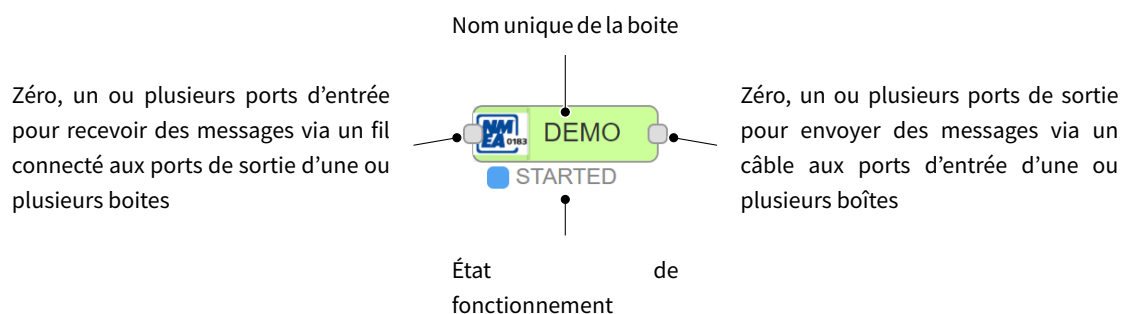
Les boîtes de *liens* permettent d'envoyer des messages d'un flux à un autre.



Les boutons  permettent de modifier le zoom.

## 6.5 boîte Manta

Une boîte est un bloc fonctionnel indépendant couvrant une fonction d'acquisition, de traitement ou d'exportation de données.

**Vue d'ensemble :****Fig. 24 :** Vue d'ensemble d'une boîte Manta

L'option *Show box status* doit être sélectionnée dans le menu Manta pour afficher l'état des boîtes

Une boîte peut représenter :

- Une entrée/sortie physique (port série, CAN, Ethernet ...)
- Un équipement avec lequel communiquer (système de navigation, batterie, éolienne...)
- Un logiciel tiers avec lequel communiquer (événements, cartes...)
- Un protocole à encoder/décoder (NMEA0183, NMEA2000...)
- Un traitement mathématique (filtrage, calibration...)
- Une fonction interne (lien, sonde, datalogger...)
- Un algorithme spécifique à l'utilisateur écrit avec Python 2.7 (y compris la bibliothèque Numpy).
- Une interaction utilisateur associée à un widget de tableau de bord de l'application Web Exocet (bouton, scalaire...)
- ...

Une boîte est constituée de :

- Un *nom unique* pour l'identifier.
- Un ensemble de *Ports d'entrée* pour recevoir des messages d'une ou plusieurs boîtes
- Un ensemble de *Ports de sortie* pour envoyer des messages vers une ou plusieurs boîtes
- Un ensemble de *Propriétés* qui correspond aux paramètres de réglage des boîtes
- Un *Etat* pour indiquer l'état interne de la boîte

Chaque port peut être associé à :

- Un *Label* pour le nommer ou le renommer.
- Un *Timeout* pour passer en *avertissement* lorsqu'aucun message n'est reçu ou envoyé.

**Conditions d'exécution :**

- A la réception de données via une I/O physique de l'Exocet
- A la réception d'un message sur un port d'entrée de la boîte
- Périodiquement sur timer

**Documentation :**

En cliquant sur une boîte, une aide contextuelle est visible dans le panneau d'information. Cette aide contient généralement les rubriques suivantes :

- Description : description générale du fonctionnement de la boîte
- Propriétés : description des paramètres
- Entrées : description des données reçues via les ports d'entrée
- Ouputs : description des données transmises via les ports de sortie



L'option *Show sidebar* doit être sélectionnée dans le menu Manta pour afficher le panneau d'information.

**Les paramètres :**

**Edit analogue box**

Delete Cancel Done

▼ **Box properties**

Name analogue

Port Ain-1

Range 0v/5V

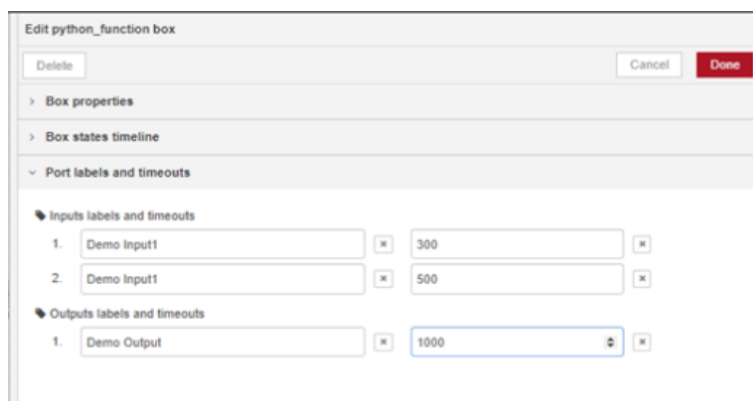
Sampling Rate (Hz) 1

Add TS ☐

**FIG. 25 :** Paramètres de la boîte Manta

Le panneau d'édition est accessible en double-cliquant sur une boîte. L'onglet **Propriétés de la boîte** de ce panneau permet de modifier le nom et les paramètres de la boîte.

La liste des paramètres est spécifique à chaque boîte. Si nécessaire, le panneau d'information fournit une description de chacun des paramètres.



**FIG. 26 :** Libellés et délais d'attente des ports

L'onglet *Port labels and timeout* du panneau d'édition peut être utilisé pour :

- Changer le label associé à chaque port.
- Associer un timeout en millisecondes.
- Pour un port d'entrée : la boîte passe à l'état WARNING si aucun message n'est reçu à la fin du timeout.
- Pour un port de sortie : la boîte passe à l'état WARNING si aucun message n'est émis à l'issue du timeout.



L'icône  est visible si un timeout a été associé à un port.

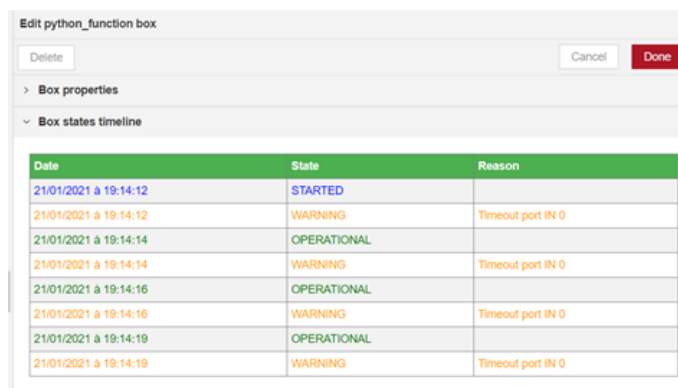
### Les états :

Pendant son fonctionnement, une boîte peut prendre les états suivants :

- **STARTED** : La boîte est correctement initialisée, mais elle n'a pas encore reçu ou envoyé de messages.
- **OPERATIONAL** : La boîte a reçu ou transmis avec succès au moins un message
- **WARNING** : Délai d'attente déclenché par un port ou un dysfonctionnement interne mineur. Cet état est toujours accompagné d'une description de la raison du déclenchement
- **ERROR** : Dysfonctionnement majeur. Cet état est toujours accompagné d'une description de la raison du déclenchement. La raison « revision mismatch, please replace this box » indique

que la boîte doit être remplacée car sa version n'est plus compatible avec la version actuelle du firmware.

La chronologie des changements d'état d'une boîte est visible dans l'onglet Box states timeline du panneau d'édition.



Date	State	Reason
21/01/2021 à 19:14:12	STARTED	
21/01/2021 à 19:14:12	WARNING	Timeout port IN 0
21/01/2021 à 19:14:14	OPERATIONAL	
21/01/2021 à 19:14:14	WARNING	Timeout port IN 0
21/01/2021 à 19:14:16	OPERATIONAL	
21/01/2021 à 19:14:16	WARNING	Timeout port IN 0
21/01/2021 à 19:14:19	OPERATIONAL	
21/01/2021 à 19:14:19	WARNING	Timeout port IN 0

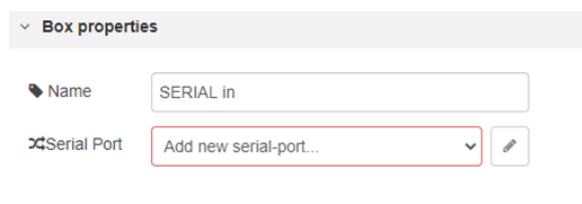
**FIG. 27 :** Chronologie des états d'une boîte Manta

## 6.6 boîte de configuration Manta



Il s'agit d'une boîte non visible qui héberge une configuration réutilisable et qui sera partagée par plusieurs boîtes.

Utilisée pour la configuration de :

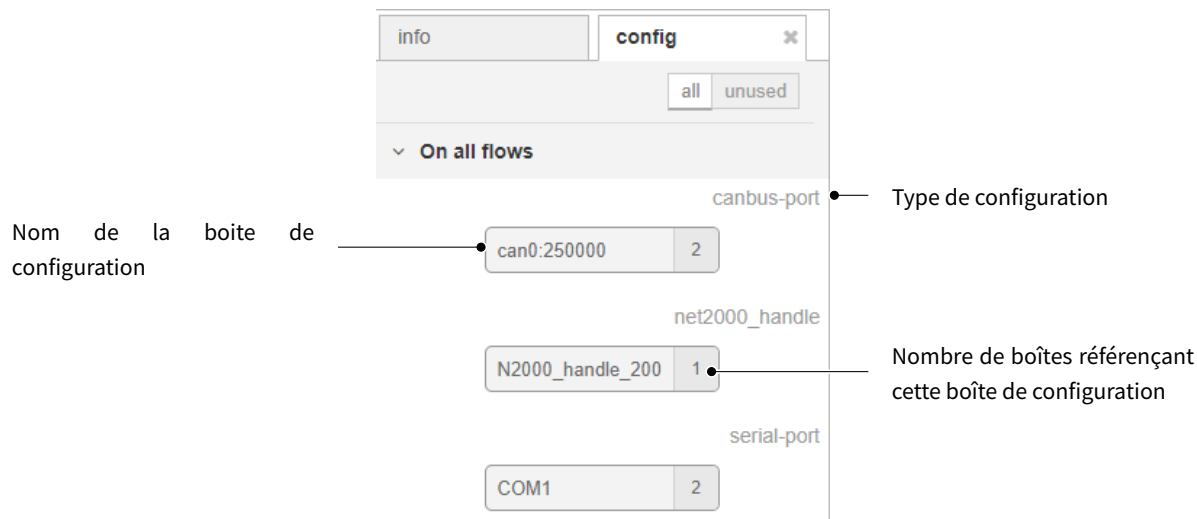
- Ports série
- Bus CAN
- Gestionnaire NMEA2000



**FIG. 28 :** Panneau de configuration de la boîte série

Dans l'onglet de propriétés d'une boîte, l'icône  indique que le paramètre correspond à une boîte de configuration. Il est alors possible de sélectionner une configuration existante dans la liste déroulante ou de créer une nouvelle configuration en cliquant sur l'icône .

Le panneau de gestion des boîtes de configuration est accessible par le menu *Manta->boites de configuration*.



**Fig. 29 :** Boîtes de configuration



Les boîtes de configuration inutilisées génèrent une charge de calcul et de mémoire inutile. Elles doivent donc être supprimées.

## 6.7 Le déploiement

Appuyer sur le bouton de déploiement  déclenche :

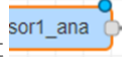
- Sauvegarde du graphe Manta dans Exocet
- Redémarrage de l'application Exocet

Si le bouton est rouge, cela signifie qu'au moins un changement a eu lieu depuis le dernier déploiement. Cela peut être dû à :

- L'ajout / suppression d'une boîte
- L'ajout / suppression d'un fil
- La suppression d'une boîte de configuration
- La modification d'un paramètre d'une boîte
- Le déplacement d'une boîte



Les boîtes qui ont été modifiées depuis le dernier déploiement affichent un cercle bleu dans leur coin supérieur droit



## 6.8 Les messages

Un **message** est une information transmise entre deux boîtes via le fil reliant un port de sortie et un port d'entrée.

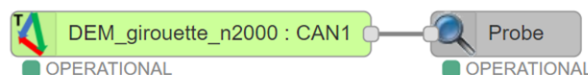
Le contenu du message est un dictionnaire qui associe le nom des variables à leur valeur, leur type et d'autres propriétés. Les messages ne sont pas nécessairement de fréquence fixe.

Types de données :

- *Booléen* : vrai ou faux
- *Integer* sur 4 octets
- *Floating* sur 4 octets
- *String* : longueur maximale de 32 caractères
- *CanFrame* : id : 4 octets, dlc : 1 octet, données : 8 octets
- *Opaque* : Buffer de 2048 octets simples (utilisé pour les données série, UDP ou TCP...)
- *FloatArray* : Buffer de 1024 éléments flottants
- *FloatTable* : Buffer de 32x32 éléments flottants (utilisé pour les tables d'upwash ou polaires...)

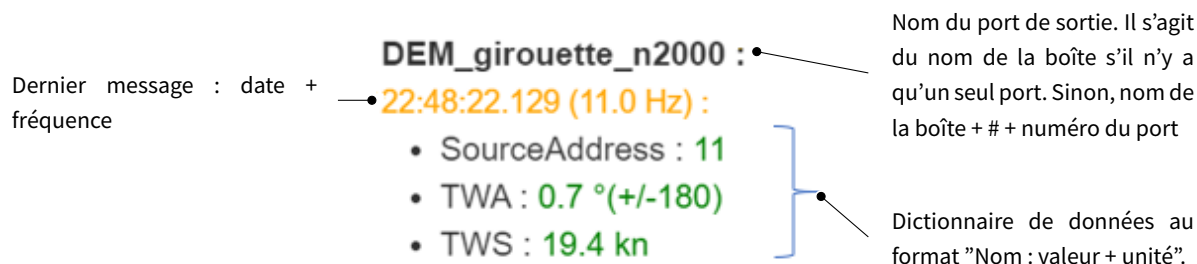
### Visualisation basique avec une sonde :

La boîte Probe affiche le message envoyé par un ou plusieurs ports de sortie



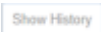
**FIG. 30 :** boîte sonde Manta

Les informations sur les messages sont disponibles en double cliquant sur la boîte sonde :

**FIG. 31 :** Informations sur les messages

Une boîte Probe peut être connectée à plusieurs ports de sortie

### Visualisation avancée avec la sonde :

Le mode  est utilisé pour afficher l'historique des messages reçus. Dans ce mode, une fois l'historique mis en pause avec le bouton, il est possible de télécharger un fichier journal des données CSV ou JSON.

Afin de ne pas pénaliser les performances du navigateur web, le rafraîchissement des données est limité à 300 ms (3,3 Hz).

Si la fréquence des données est supérieure à 3,3 Hz, vous devez utiliser l'option de vitesse maximale afin que toutes les données soient disponibles dans les fichiers CSV ou JSON.



Les sondes sont les seules boîtes dont le nom ne doit pas être unique



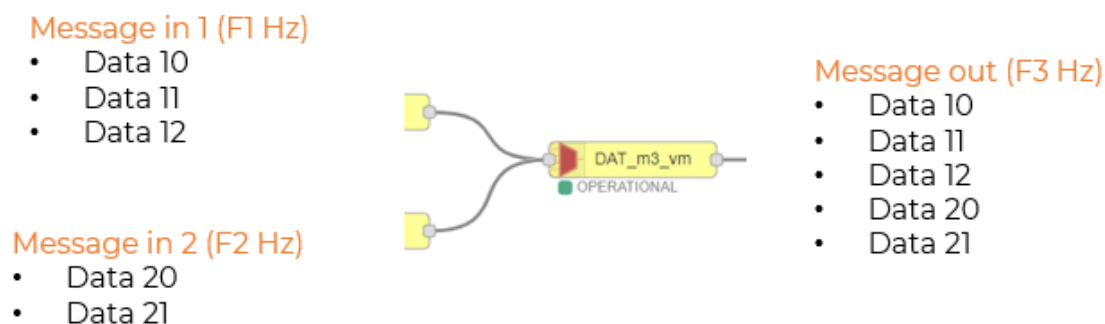
Une sonde avec l'option *full speed* utilise beaucoup de puissance de calcul. Son utilisation doit être réservée aux phases de développement. Cette option ne doit en aucun cas être utilisée dans un système en fonctionnement.

## 6.9 Les manipulations de messages

### Concaténation :

La boîte *VarMux* concatène deux messages



**FIG. 32 :** boîte VarMux

La boîte offre 3 options pour contrôler la fréquence du message sortant :

- *Fastest* : message envoyé à chaque fois qu'un message est reçu sur le port d'entrée
- *Slowest* : message transmis à la fréquence du message d'entrée le plus lent
- *Periodically*, de 0,1 à 50 Hz



L'option *fastest* peut générer une fréquence de message élevée en raison de la désynchronisation des messages reçus.



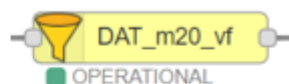
Assurez-vous qu'il n'y a pas de données portant le même nom dans les dictionnaires reçus par la boîte VarMux. Sinon, certaines données seront perdues une fois la concaténation terminée.

### Filtrage de variables :

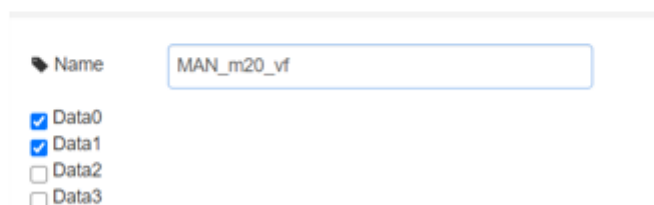
La boîte *VarFilter* supprime les variables du dictionnaire

**Message in (F Hz)**

- Data 0
- Data 1
- Data 2
- Data 3

**Message out (F Hz)**

- Data 0
- Data 1

**FIG. 33 :** boîte VarFilter

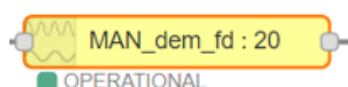
La fréquence du message de sortie est la même que celle du message d'entrée

**Sous-échantillonnage :**

La boîte *freq\_divider* permet de diminuer la fréquence d'un message

**Message in (FREQ Hz)**

- Data 0
- Data 1
- Data 2

**Message out (FREQ/20 Hz)**

- Data 0
- Data 1
- Data 2

**FIG. 34 :** boîte freq\_divider

La boîte *VarMux* peut également être utilisée pour diminuer la fréquence, surtout si la fréquence souhaitée n'est pas un facteur de la fréquence du message d'entrée.



La boîte n'applique aucune fonction d'interpolation, elle sélectionne simplement un message

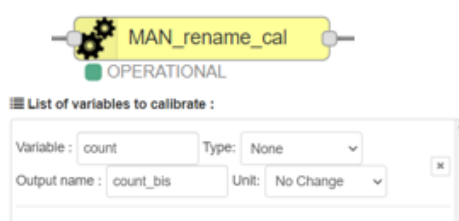
parmi N.

### Renommer des variables :

La boîte *Calibration* peut être utilisée pour renommer certaines variables d'un message.

Message in (FREQ Hz)

- count



Message out (FREQ Hz)

- Count\_bis

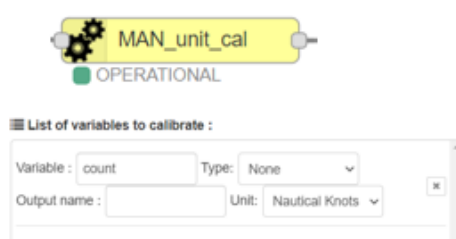
FIG. 35 : Renommer des variables

### Modification de l'unité :

La boîte *Calibration* permet de spécifier ou de modifier l'unité d'une variable.

Message in (FREQ Hz)

- Count: val



Message in (FREQ Hz)

- Count: val Kn

FIG. 36 : Modifier l'unité

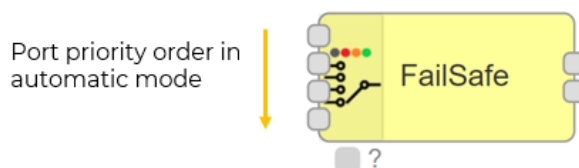
### Sélection des messages redondants :

La boîte *FailSafe* gère plusieurs entrées redondantes.

En mode automatique, le port valide le plus prioritaire est sélectionné. Si aucune donnée n'est reçue à l'échéance du délai ou si les données sont détectées comme invalides, l'entrée de priorité inférieure sera sélectionnée. L'entrée la plus prioritaire sera à nouveau sélectionnée si la réception des données reprend ou si les données retrouvent un statut valide.

À tout moment, l'utilisateur a la possibilité de forcer la sélection d'un port.

Si l'option *interrupteur* est sélectionnée dans le panneau de configuration, seule la sélection manuelle du port est disponible.

**FIG. 37 :** boîte FailSafe

Les délais d'expiration des ports sont définis dans l'onglet Étiquettes de port et délais d'expiration

Si les noms des variables des messages reçus diffèrent, il est alors possible de configurer le renommage de ces variables afin d'avoir un dictionnaire de sortie avec des noms de variables identiques, quelle que soit l'entrée sélectionnée.

#### Message anglais (0.1 Hz)

- Angle : 2 °
- Speed : 1 Kn

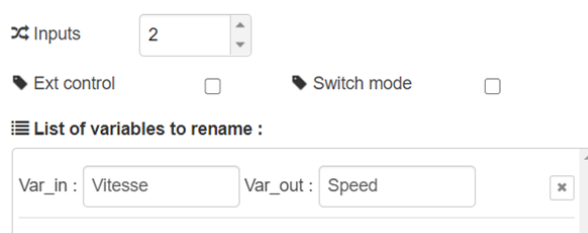
#### Message français (1 Hz)

- Angle : 200 °
- Vitesse : 50 Kn



#### Message (1 Hz)

- Angle : 200 °
- Speed : 50 Kn

**FIG. 38 :** Renommer les variables de la boîte FailSafe**FIG. 39 :** Renommer Vitesse en Speed

### Échange entre les flux :

Les boîtes *Link\_in* et *link\_out* sont utilisées pour envoyer un message d'un flux à un autre. Elles

peuvent également être utilisées pour envoyer un message à l'intérieur d'un flux sans alourdir le graphe.



En cliquant sur un lien, une étiquette indique dans quel flux se trouve le lien connecté. Vous pouvez accéder au flux connecté en cliquant sur l'étiquette.

### Échange entre exocets :

Les boîtes *exocet\_in* et *exocet\_out* sont utilisées pour envoyer des messages d'un Exocet à un autre. Pour éviter tout conflit de noms, les données sont renommées avec le format BOXNAME\_VARIABLENAME avant la transmission.

### Messages de maintenance :

La boîte *Système* exporte les données de maintenance suivantes :

- Charge des bus CAN
- Charge des CPU
- Occupation de la mémoire RAM
- Occupation de l'espace de la carte SD
- Durée de fonctionnement
- Température du processeur



Une boîte *Système* doit être présente dans le graphique Manta pour que ces données soient visibles dans la page *Status* de l'application Web.

La boîte *manta\_status* informe sur le nombre et le nom des boîtes Manta en état WARNING ou ERROR. Cette information peut être enregistrée comme donnée ou pour une notification de l'utilisateur.

## 6.10 Traitement Manta

Manta fournit un certain nombre de boîtes de traitement de données :

- La boîte *calibration* permet de renommer une variable, de changer son unité, ou d'ajuster la précision et l'exactitude de sa valeur. Pour ce faire, différentes méthodes d'ajustement sont disponibles : par fonction affine ou par table de correction (LUT).
- La boîte *moving\_mean* effectue des calculs glissants (moyenne, variance, médiane, min / max...) pour une liste de variables.
- La boîte *damping* applique un filtrage.

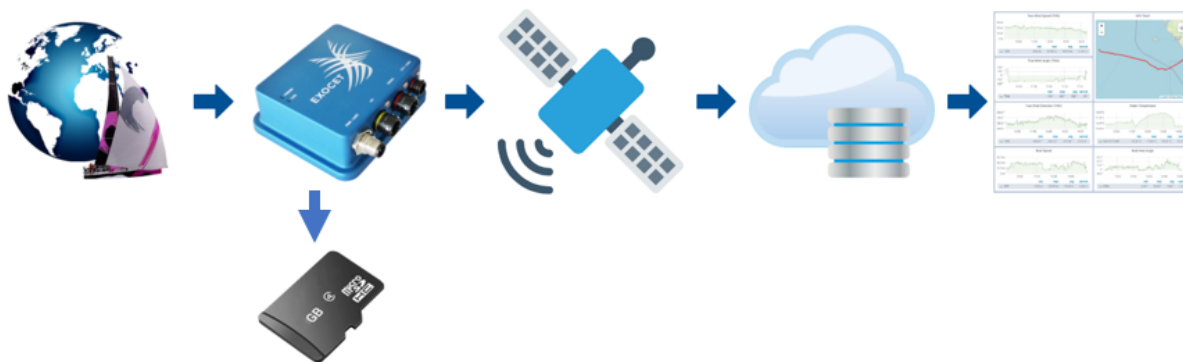
- La boîte *delay* applique un retard pour synchroniser précisément plusieurs messages.
- La boîte *expression* permet d'appliquer toutes les opérations arithmétiques fondamentales et des branchements conditionnels.



La boîte *calibration* peut être associée à un widget *courbe* pour visualiser la fonction d'ajustement dans un tableau de bord.

## 6.11 Enregistrement de données

A réception par la boîte *datalogger*, les données sont stockées sur une carte SD. En option, elles peuvent être transmises à notre application de gestion des données dans le cloud <https://exocet.cloud/>.



**FIG. 40 :** Enregistrement des données



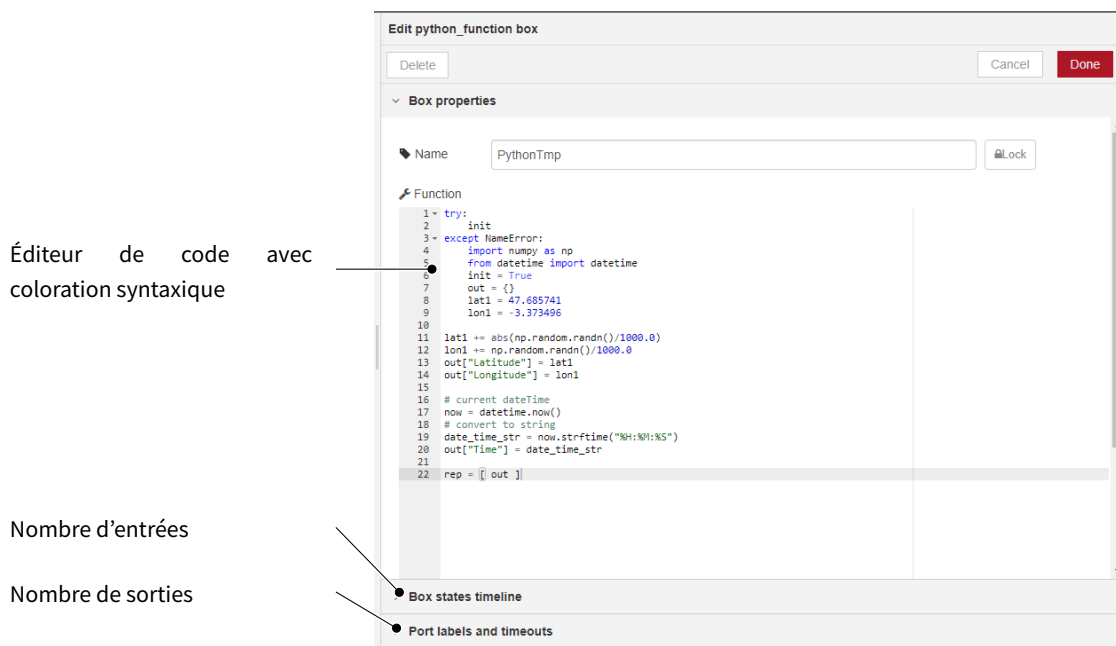
Cette fonctionnalité est spécifique à Exocet Blue.



Pour pallier à d'éventuels conflits de noms de données, celles-ci sont renommées avec le format `BOXNAME_VARIABLENAME` avant l'enregistrement ou la transmission vers le cloud...

## 6.12 Scripting Python

La boîte *python\_function* offre un interpréteur Python 2.7 accompagné d'un large choix de bibliothèques, dont NumPy.

**FIG. 41 :** Editeur python**Principe général de fonctionnement :**

Le script est exécuté chaque fois que des données sont reçues par un port d'entrée. Après l'exécution du script, des messages peuvent être envoyés depuis les ports de sortie.



La boîte *tick* peut être utilisée pour cadencer l'exécution d'un script qui ne reçoit pas de données externes.

**Variables Python spéciales :**

- *msg* : Dictionnaire contenant les données du port qui a déclenché l'appel au script
- *port* : Index du port qui a déclenché l'appel au script.
- *rep* : Tableau des dictionnaires permettant d'exporter les données hors de la boîte. La taille du tableau = nombre de ports de sortie. Chaque élément du tableau est soit un dictionnaire, soit None. Si l'élément est None, aucune donnée ne sera sortie du port correspondant.

**Les étapes du fonctionnement :**

1. Un message est reçu par l'un des ports d'entrée de la boîte *python\_function*
2. Attente de la disponibilité de l'interpréteur
3. Chargement du contexte local du script

4. Création de la variable *msg* à partir du message reçu. Les variables Manta sont converties en variables Python.
5. Assignment de l'index du port d'entrée dans la variable *port*.
6. Exécution du script Python
7. Création des messages qui seront envoyés par les ports de sortie en fonction de la variable *rep*. Les variables Python sont converties en variables Manta.
8. Sauvegarde du contexte local
9. Libération de l'accès à l'interpréteur
10. Envoi des messages via les ports de sortie

### Hello World :

```
out={} ;
out["hello"] = "Hello World!";
rep = [ out ] ;
```

### Conversion de type Manta vers Python :

Manta	Python
Booléen, flottant, entier, chaîne de caractères	Équivalent Python
Opaque	ByteArray
Array, Table	FloatTable
CanFrame	pixel.CanFrame

La classe `pixel.CanFrame` est composée des champs :

- Id : nombre entier pour l'ID CAN
- Bytes : ByteArray de données

### Conversion de type Python vers Manta :

Les données Python sont converties en données Manta lors de la création des messages à envoyer aux ports de sortie. Le tableau python *rep* contient la liste des messages à transmettre.

Python	Manta
Booléen, flottant, entier	Équivalent Manta
String	String tronqué à 32 caractères maximum



Python	Manta
ByteArray	Opaque tronqué à 2048 octets
FloatTable	Tableau limité à 1024 flottants, ou tableau limité à 32x32 flottants
pixel.CanFrame	CanFrame

**Attention :**

Il n'y a qu'un seul interpréteur Python. Lorsqu'une boîte exécute son script, les autres sont bloquées, en attente de la disponibilité de l'interpréteur. Afin de ne pas pénaliser les performances générales du système, il est donc conseillé de limiter au maximum :

- La complexité des scripts
- La fréquence des messages envoyés aux boîtes Python. 10Hz maximum.


L'utilisation des boîtes Python doit être réservée aux traitements qui ne peuvent être effectués par une autre boîte du catalogue (ex : boîte Expression).

**6.13 Bibliothèques Manta**

Les bibliothèques sont utilisées pour importer ou exporter tout ou partie d'un flux.

**Créer une bibliothèque :**

Depuis Manta :

1. Sélectionnez un ensemble de boîtes avec la souris ou toutes les boîtes d'un flux avec CTRL + A. Depuis le menu, sélectionnez *Export->Library*.
2. Donnez un nom à la bibliothèque puis appuyez sur le bouton 

**Instaurer une bibliothèque dans un flux :**


Depuis Manta :

1. Dans le menu, sélectionnez Import-> Library-> Nom de la bibliothèque
2. Cliquez dans le flux à l'endroit souhaité pour placer le contenu de la bibliothèque.



Si la bibliothèque contient des boîtes de liens, celles-ci doivent être recréées.

## Gestion de la bibliothèque :

L'interface de gestion de la bibliothèque est accessible via le bouton  de la page *Statut* de l'application web.



**FIG. 42 :** Panneau de gestion des bibliothèques Manta

## 6.14 Les bonnes pratiques de Manta

Pour garder un graphique Manta clair qui permet de faciliter la maintenance et l'évolution, et pour éviter une charge CPU inutile, quelques bonnes pratiques ont été identifiées :

### Organiser le graphe Manta :

- Séparer les processus fonctionnels en flux pertinents de taille modeste.
- Utiliser des boîtes *Lien* à l'intérieur d'un flux pour éviter les croisements de fils.
- Définir et respecter une convention de nommage pour les noms de boîtes et de flux.
- Donner un nom aux boîtes *Lien* pour une meilleure compréhension des connexions entre les boîtes.



### Erreurs à éviter :

- Ne pas envoyer de variables angulaires sans unité aux boîtes *damping*, *moving\_mean* et *calibration*.
- Une boîte avec option de transmission périodique utilisée avant une boîte *failsafe* empêche la boîte de détecter la perte de réception des données.
- Lors de l'instanciation d'une bibliothèque Manta, les boîtes *Link in* et *out* doivent être remplacées.
- Evitez que plusieurs utilisateurs modifient le graphe Manta en même temps.
- N'oubliez pas de sauvegarder la configuration après une mise à jour, en utilisant le bouton *Exporter la configuration* de la page *paramétrage* de l'application Web.

**Pour limiter la charge CPU et l'occupation mémoire :**

- Limitez la fréquence des messages trop rapides le plus tôt possible dans le graphique, notamment ceux qui seront envoyés à une boîte Python.
- La fréquence sera contrôlée par la boîte *Probe*.
- La limitation de la fréquence se fera avec les boîtes *freq\_divider* ou *VarMux*.
- Limiter la complexité et le nombre de scripts *Python*.
- Filtrer les variables dès que possible pour limiter le nombre de variables inutiles dans les messages avec une boîte *varfilter*.
- Limiter au maximum le nombre de *sondes*. Pas de *probe* en mode full-speed en dehors de la phase de débogage.
- Supprimer les boîtes de configuration non référencées par une boîte.

**Pour vérifier le bon fonctionnement :**

- Vérifier l'état des boîtes depuis la barre de navigation de l'application web .
- Utilisez la chronologie des boîtes dans l'état WARNING ou ERROR pour obtenir des informations sur le moment et la raison du problème.
- Vérifiez la charge CPU et l'occupation de la mémoire à partir de la page *Status* de l'application web. Pour un système sain :
  - Total des CPU et chaque CPU < 80 %
  - Mémoire RAM < 80%
  - Température < 80°
- Vérifiez le temps de fonctionnement à partir de la page *status/Système*. Ce temps est remis à zéro en cas de redémarrage intempestif.
- En cas d'anomalie, sauvegarder les logs du système à l'aide du bouton  de la page *status* pour analyse par Pixel Sur Mer.

**Nommage des flows :**

Voici une convention utilisée pour nommer les flows : **[prefix]\_[TRIGRAM]restofthename**

- Selon le traitement majoritaire réalisé par le flow, le préfixe doit être du type suivant :

Prefix | Treatment | Préfixe | Traitement | |----|-----| | in | Entrées de capteur ou réception de données | | out | Sortie vers actionneurs, bus ou visualisation | | pro | Calcul ou traitement de donnée | | wnd | Calcul de vent | | ovl | Surcouche de pilotage | | plt | Communication de Pilot overlay avec un pilote | | simu | Simulation |

- Le trigramme est composé de 3 ou 4 lettre majuscule et doit correspondre au nom de la fonction majoritaire du flow. (ex : INS, MAST, etc..). C'est ce trigramme qui sera repris dans le nommage des éléments du flow.

- La suite du nom est en minuscule. Elle doit être la plus courte possible et complète au besoin le nom du flow pour plus de clarté.

### Découpage flows :

Flows de référence pré-définis, standards et fournissant un dictionnaire établi de données :

Nom	Fonction
in_INS	Réception et traitement des données inertielles
in_BSP	Réception et traitement de la vitesse bateau
in_GRE	Réception et traitement des données de charge de grément
in_NRJ	Réception des données d'énergie : batterie, hydrogéné, panneau solaire, ...
in_MAST	Réception et traitement des données du mat
in_MHU	Réception et traitement des données du MHU
in_GPS	Réception et traitement des données GPS
in_out_HMI	Interface homme machine (clavier blink, afficheur)
wnd_LEEWay	Calculs de dérive
wnd_WIND	Calcul de vent
out_LOG	Fourniture de log vers enregistrements
out_ToH5K	Fourniture de données vers équipement H5K
plt_BAG	Dialogue avec pilote B and G
plt_NKE	Dialogue avec pilote NKE
ovl_MNG	Calcul du Manager de surcouche
ovl_HSafety	Calcul de la surcouche heel safety
ovl_HRegulation	Calcul de la surcouche heel regulation
ovl_AWARegulation	Calcul de la surcouche awa regulation
ovl_SPDRegulation	Calcul de la surcouche speed regulation
ovl_VMGPerf	Calcul de la surcouche VMG performance

Cas des flows contenant des boîtes d'entrées (ex : Analog, CAN, Series, UDP, etc ..) : Si les données

concernées seront utilisées par plusieurs fonctions, le traitement de leur réception doit être mis dans flow dédié (préfixé avec « in\_ »). Ce flow contiendra alors le traitement de la calibration, des failsafe et des éventuels filtrages. Dans le cas contraire, il est conseillé de gérer l'ensemble du traitement (et des traitements associés) dans le même flow. Il sera alors nommé en rapport avec la fonction réalisée.

### Nommage des liens « out » entre flows

Voici une convention utilisée pour nommer les liens out : **NomExpliciteDictionnaire**

Utilisation de la notation Camel Case avec un nom représentatif des données exportées contenues dans son dictionnaire. Remarque : Son nom ne peut pas être issu de la destination prévue pour cette boîte car celle-ci peut être liée à plusieurs boîtes « in » destinataire

### Nommage des liens « in » entre flows

Utilisation de la notation Camel Case en distinguant 2 cas :

- Si la boîte « In » est en lien dans le flow avec plusieurs fonctions, alors son nom doit être représentatif des données importées contenues dans son dictionnaire : **NomExpliciteDictionnaire**
- Si la boîte « In » n'est en lien dans le flow qu'avec une fonction unique, alors son nom peut être représentatif de la boîte destination : **NomExpliciteDestination**

### Nommage des boîtes de fonction

Voici une convention utilisée pour nommer les boîtes : **\*\*[TRIGRAMME flow]\_[NomCamelCase]\_[suffixe type de fonction]\*\***

- On rappelle le trigramme du flow contenant la boîte
- On associe le nom de la donnée, du traitement ou du matériel concernée par la boîte
- On ajoute un suffixe (optionnel) indiquant le type de calcul réalisé parmi les types suivants :

Suffixe	Traitement
ana	réception analogique
cal	Calibration
py	Python
damp	Filtrage Temporel
mov	Calcul sur fenêtre glissante (moyenne...)
vf	Filtrage des variables dans un dictionnaire
fs	Fonction FailSafe
sel	Boîte failsafe utilisée comme sélecteur

Suffixe	Traitement
pid	Régulateur PID
btn	Boite Bouton
mux	Boite utilisée pour multiplexer plusieurs dictionnaire
udpIn	Lecture port protocole UDP
udpOut	Ecriture port protocole UDP
n2kIn	Lecture protocole NMEA2000
canIn	lecture interface CAN
canOut	Ecriture interface CAN
serialIn	Lecture port Serie
serialOut	Ecriture port Serie
tk	Boite tick

### Nommages des variables

Here is a convention used to name variables : **nomExpliciteCamelcase\_[suffixe éventuel]**

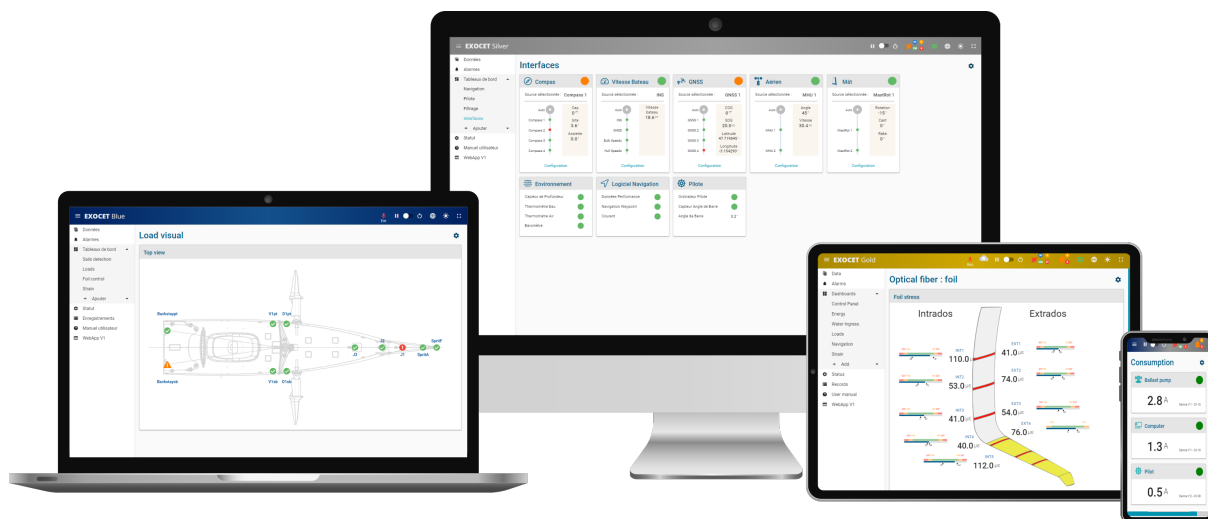
- Utilisation de la notation Camel Case avec un nom représentatif de la donnée concernée.
- Éventuellement suffixe pour précision : \_raw, etc...
- Les variables suivantes possèdent un nom standard et correspondent aux données suivantes :

Nom	Unités	Description
AWA	°	Angle vent apparent / bateau
AWS	kn	Vitesse vent apparent
TWA	°	Angle vent réel / bateau
TWD	°	Direction vent réel (vent / eau)
TWS	kn	Vitesse vent réel (vent / eau)
MastRot	°	Angle mat selon l'axe vertical / bateau
MastRake	°	Angle mat selon l'axe transversal / bateau
MastCant	°	Angle mat selon l'axe longitudinal / bateau
BoatSpeed	kn	Vitesse longitudinale bateau / eau

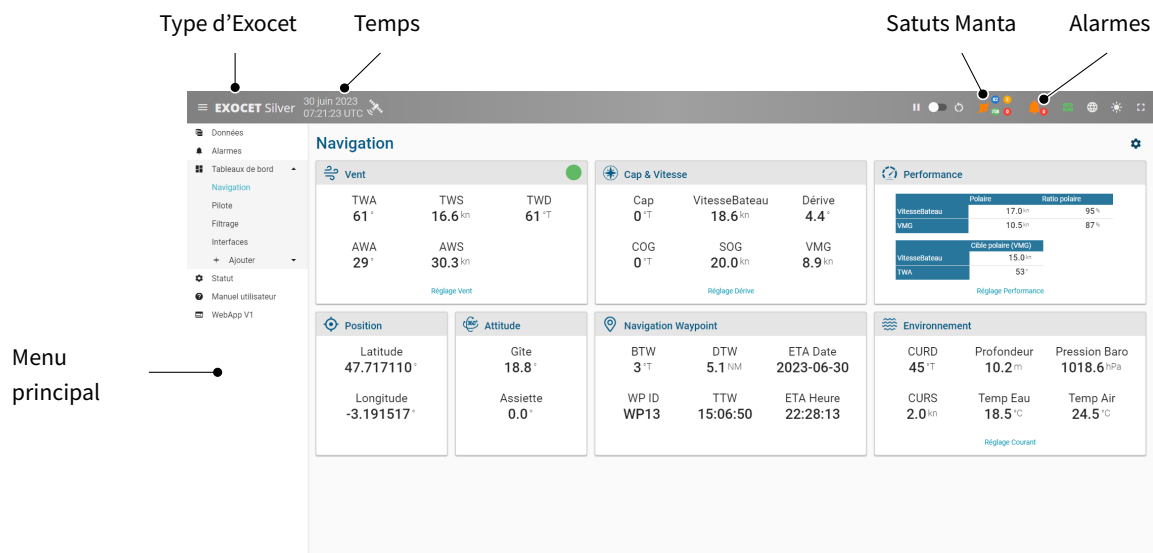
Nom	Unités	Description
Leeway	°	Angle dérive / eau
Course	°	Direction vitesse bateau / eau
SurgeVel	m/s	Vitesse longitudinale bateau / terre
SwayVel	m/s	Vitesse latérale bateau / terre
HeaveVel	m/s	Vitesse verticale bateau / terre
SurgeAcc	m/s <sup>2</sup>	Accélération longitudinale bateau / terre
SwayAcc	m/s <sup>2</sup>	Accélération latérale bateau / terre
SOG	kn	Vitesse bateau / terre
COG	°	Direction vitesse bateau / terre
Lat	°	Position GPS latitude
Long	°	Position GPS longitude
Heading	°	Direction bateau / nord
Trim	°	Angle bateau selon l'axe transversal / terre
Heel	°	Angle bateau selon l'axe longitudinal / terre
YawRate	°/s	Vitesse de rotation bateau selon l'axe vertical / terre
PitchRate	°/s	Vitesse de rotation bateau selon l'axe transversal / terre
RollRate	°/s	Vitesse de rotation bateau selon l'axe longitudinal / terre
CURD	°	Direction courant
CURS	kn	Vitesse courant

## 7 Application Web Exocet

### 7.1 Aperçu de l'application Web Exocet



**FIG. 43 :** Interfaces utilisateur multi-écrans



**FIG. 44 :** Page web Exocet

L'Exocet Web App est une interface utilisateur logicielle accessible via tous les supports mobiles et tous les navigateurs web. Elle est composée d'une barre de navigation en haut où l'on trouve :



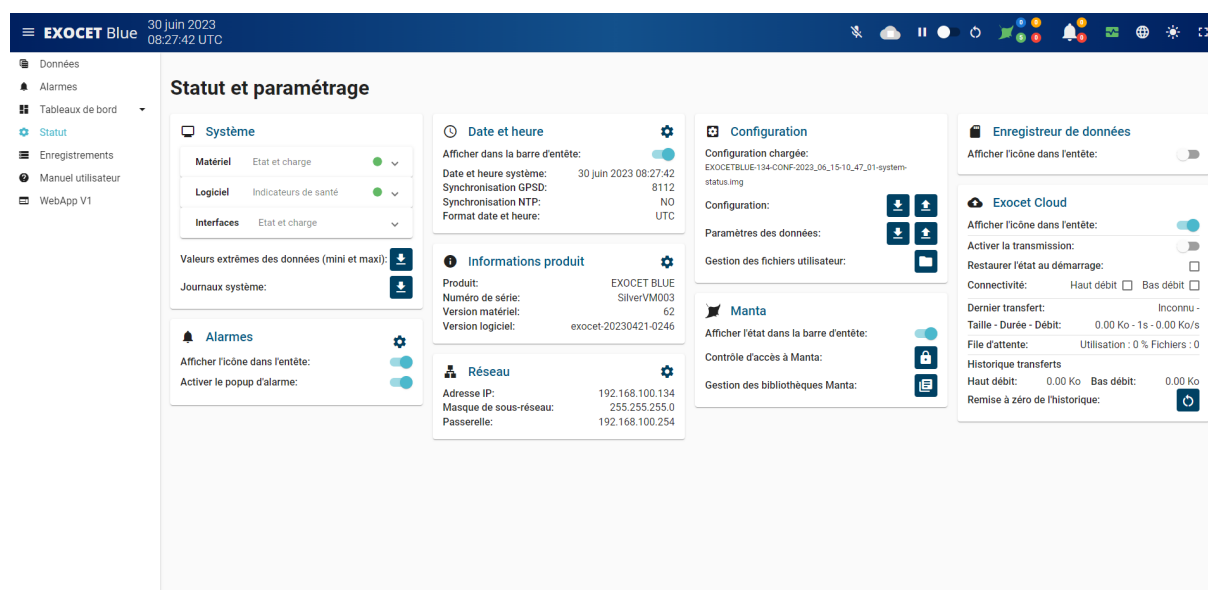
- Un statut visuel des boîtes Manta, avec un lien vers l'espace de travail Manta.
- Un statut visuel des alarmes.
- Un indicateur de date/heure synchronisé

Le menu principal, à gauche, offre plusieurs liens vers d'autres pages :

- *Données* mène à la page Table de données.
- *Alarmes* dirige vers la page de gestion des alarmes.
- *Tableaux de bord* liste sous forme de menu déroulant les tableaux disponibles sur le produit.
- *Statut* renvoie sur une page de configuration générale de l'Exocet
- *Manuel utilisateur* télécharge la dernière version du manuel utilisateur (ce document)
- *WebApp V1* permet d'accéder à l'ancienne version web des produits Exocet

Au centre, le contenu dépend de la page sélectionnée.

## 7.2 Page « Statut » de l'application Web Exocet



**Fig. 45 :** Page de statut et paramétrage de l'application web Exocet

La page de Statut est utilisée pour configurer :

- Interface utilisateur : langue, date/heure, mode d'affichage nuit/jour, la liste des variables à afficher dans le pied de page.

- Options des données : décimales par défaut pour l'affichage des données flottantes. Ces paramètres peuvent être surchargés pour une donnée en particulier à partir de la page « Table de données ».
- Paramètres du système :
- Synchronisation de l'heure et de l'horloge :
  - Appliquer l'heure du PC au système
  - Configurer le port UDP du GPSD pour la synchronisation du temps.
  - Configurer un serveur NTP distant pour la synchronisation de l'heure. Notez que tout produit Exocet peut être utilisé comme serveur Chrony.
- Exporter la configuration actuelle du système :
  - Graphique Manta
  - Liste des données à afficher en pied de page
  - Alarmes de données, alias et décimales
  - Liste des graphiques
  - Tableaux de bord
  - Réglage des décimales par défaut
  - Définition des pages mobiles
- Téléchargement d'une configuration de système
- Redémarrage de l'ensemble du système ou de l'application Manta
- Téléchargement d'un nouveau firmware (préservation de la configuration actuelle)
- Créer un contrôle d'accès Manta
- Gérer les librairies Manta
- Réinitialisation à l'état d'usine. Cela détruit la configuration du système, mais pas les paramètres du réseau ni les fichiers d'enregistrement.
- Exportation des enregistrements (Exocet Blue seulement) : Configurez la façon dont les enregistrements de données sont encodées au format CSV, et téléchargez l'installateur du logiciel PSM Decoder. Ce logiciel est obligatoire pour exporter les fichiers d'enregistrement de l'Exocet vers le PC de l'utilisateur.

On retrouve également sur cette page certains états du système :

- Date et heure du système
- Etat de la synchronisation GPSD & NTP
- Temps de fonctionnement du système
- Charges totales et détaillées du CPU
- Utilisation de la mémoire RAM
- Température du système

- Utilisation de l'espace d'enregistrement
- Utilisation du nombre de fichiers d'enregistrement
- Téléchargement des journaux du système : à fournir à Pixel Sur Mer pour l'analyse des problèmes.

Et aussi quelques informations sur le produit :

- Nom du produit
- Numéro de série du produit
- Référence du matériel
- Version du logiciel
- Nom de la dernière configuration chargée

### 7.3 Page « Enregistrements » de l'application Web Exocet

Actions	Début ↓	Fin	Durée	Taille
	28/06/2023 13:39:10	29/06/2023 00:00:00	10:20:49	68.66 Mo
	28/06/2023 13:37:20	28/06/2023 13:39:04	0:01:43	0.96 Mo
	22/06/2023 09:39:44	22/06/2023 11:24:35	1:44:51	1.52 Mo
	21/06/2023 10:00:02	21/06/2023 10:01:01	0:00:58	0.87 Mo
	21/06/2023 00:00:00	21/06/2023 09:16:53	9:16:52	61.47 Mo
	20/06/2023 00:00:00	21/06/2023 00:00:00	23:59:59	156.93 Mo
	19/06/2023 13:43:43	20/06/2023 00:00:00	10:16:17	67.77 Mo
	19/06/2023 13:31:06	19/06/2023 13:43:37	0:12:30	2.18 Mo
	19/06/2023 13:19:07	19/06/2023 13:31:00	0:11:52	2.11 Mo
	19/06/2023 08:27:01	19/06/2023 13:18:55	4:51:54	32.40 Mo
	19/06/2023 07:59:37	19/06/2023 08:26:54	0:27:17	3.15 Mo
	19/06/2023 07:49:02	19/06/2023 07:59:26	0:10:24	1.70 Mo
	19/06/2023 07:30:29	19/06/2023 07:48:50	0:18:21	2.35 Mo
	19/06/2023 00:00:00	19/06/2023 07:29:51	7:29:50	38.87 Mo
	18/06/2023 00:00:00	19/06/2023 00:00:00	23:59:59	121.64 Mo

**Fig. 46 :** Page Enregistrements de l'application web de l'Exocet

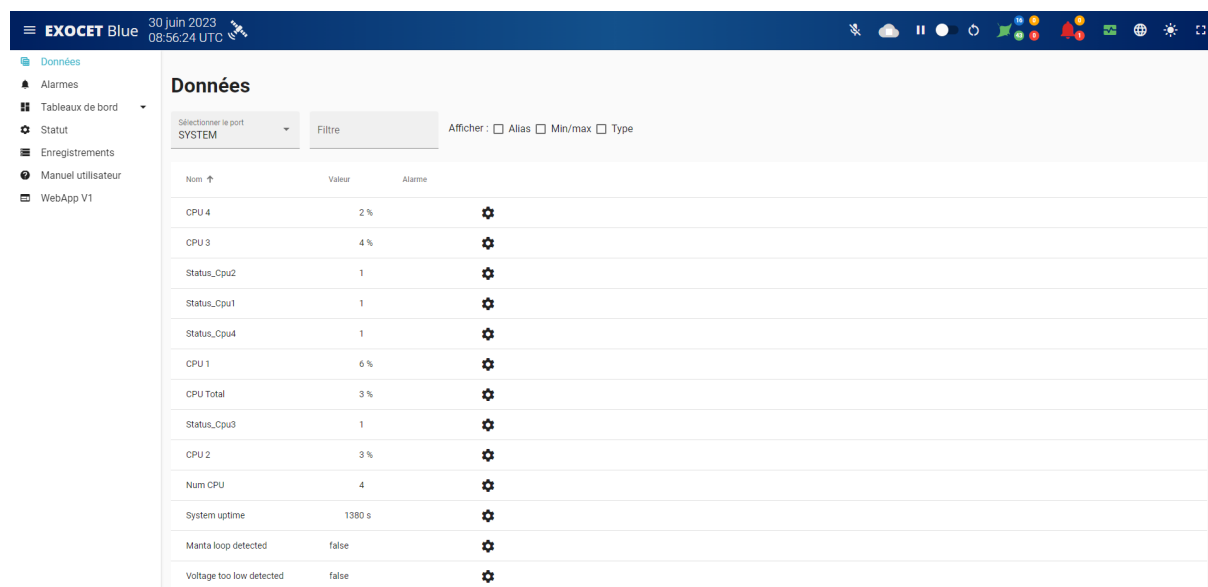
La page Enregistrements, disponible uniquement sur l'Exocet Blue, permet de gérer les fichiers d'enregistrement stockés sur l'Exocet.

Les deux boutons à gauche de chaque ligne permettent de télécharger le fichier au format PSM ou au format CSV. Le format PSM est un format optimisé. Le décodeur PSM est nécessaire pour décoder un fichier au format PSM.

Une note peut être attachée au fichier en cliquant sur la droite de la ligne.

Le bouton *Supprimer* à gauche permet de supprimer les fichiers sélectionnés.

## 7.4 Page « Table de données » de l'application Web Exocet

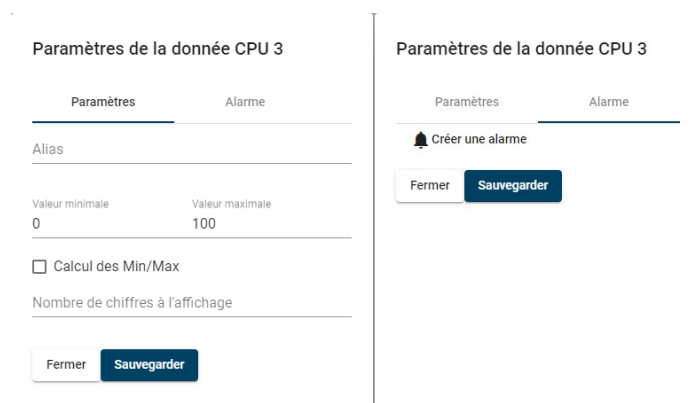


Nom	Valeur	Alarme
CPU 4	2 %	
CPU 3	4 %	
Status_Cpu2	1	
Status_Cpu1	1	
Status_Cpu4	1	
CPU 1	6 %	
CPU Total	3 %	
Status_Cpu3	1	
CPU 2	3 %	
Num CPU	4	
System uptime	1380 s	
Manta loop detected	false	
Voltage too low detected	false	

**FIG. 47 :** Page Table de données de l'application web Exocet

La page Table de données est utilisée :

- pour afficher rapidement les données de sortie de n'importe quel boîtier Manta en utilisant le bouton *Select* sur la gauche.
- d'accéder à la configuration des variables. En appuyant sur l'icône crayon, la boîte de configuration des données s'affiche.



Paramètres de la donnée CPU 3

Paramètres

Alarme

Alias

Valeur minimale

Valeur maximale

0

100

☐ Calcul des Min/Max

Nombre de chiffres à l'affichage

Fermer

Sauvegarder

Paramètres de la donnée CPU 3

Paramètres

Alarme

Créer une alarme

Fermer

Sauvegarder

**FIG. 48 :** boîte de configuration des données

Pour une donnée particulière, l'utilisateur peut définir :

- Un alias : qui sera utilisé partout où la donnée est affichée (pied de page, graphique, tableaux de bord), mais pas sur les fichiers d'enregistrement.
- Une valeur min/max : qui sera utilisée par les graphiques et certains widgets des tableaux de bord.
- Nombre de décimales à afficher : pour écraser la valeur par défaut du système pour cette donnée
- Nombre de chiffres affichés : force le nombre minimum de chiffres utilisés pour afficher cette donnée
- Une alarme : à définir avec :
  - une option d'activation
  - un niveau de sévérité
  - un seuil haut / bas pour déclencher une alarme
  - un seuil haut / bas pour déclencher un avertissement

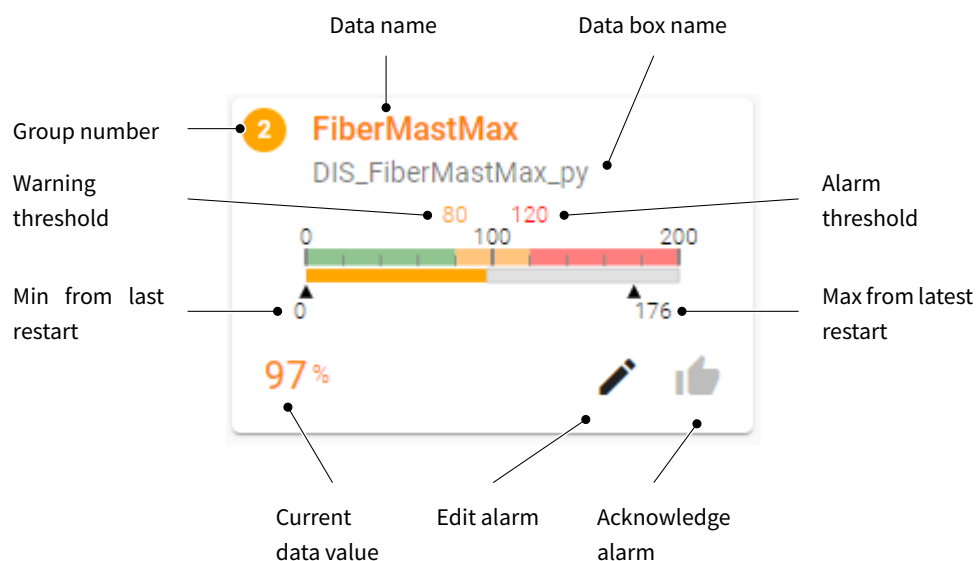
Si un seuil d'alarme est atteint, une alarme ou un avertissement sera déclenché. La page « Alarme » peut être utilisée pour surveiller les alarmes définies.

## 7.5 Page « Alarmes » de l'application Web Exocet



**Fig. 49 :** Page Alarmes de l'application web Exocet

La page des alarmes affiche une animation de jauges linéaires pour chaque donnée avec un seuil d'avertissement et/ou d'alarme défini.



**FIG. 50 :** Une jauge linéaire d'alarme

Chaque jauge indique :

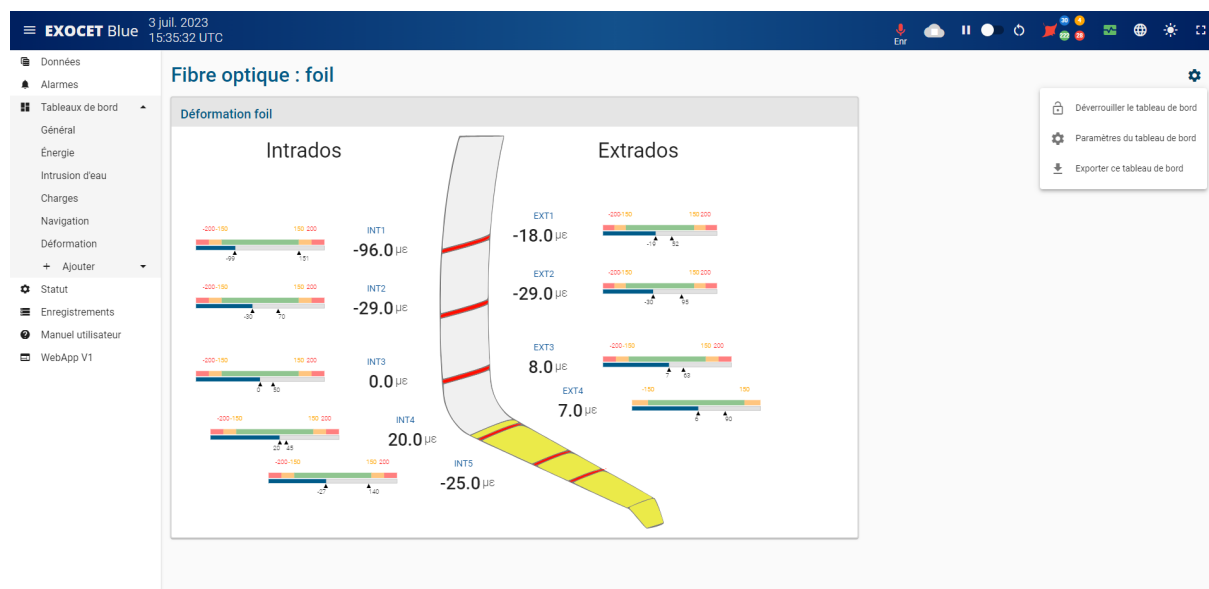
- Le niveau de sévérité de l'alarme
- le nom de la donnée et de la boîte concernée
- Les seuils d'avertissement et/ou d'alarme
- Valeur maximale atteinte depuis le démarrage du système
- La valeur actuelle de la donnée
- Un bouton d'édition pour configurer l'alarme
- Un bouton d'acquiescement pour désactiver l'alarme, disponible uniquement si les conditions requises sont respectées.

L'état des alarmes et des avertissements actifs est reporté dans la barre de navigation en haut de l'écran.

Une boîte *Alarme* peut être utilisée dans le graphique Manta pour contrôler un GPIO, et pour faire sonner un buzzer par exemple.

## 7.6 Tableaux de bord de l'application Web Exocet

### Tableau de bord



**FIG. 51 :** Aperçu des pages de tableaux de bord

Les pages de tableaux de bord sont utilisées pour afficher ou contrôler en direct des données de boîtes Manta dans un environnement convivial.

Un tableau de bord est une page qui peut afficher une image de fond et un ensemble de widgets. Les widgets sont des composants graphiques animés qui sont associés à une donnée Manta. L'emplacement des widgets en fonction de l'image de fond peut être défini et sauvegardé. L'animation graphique dépend des données.

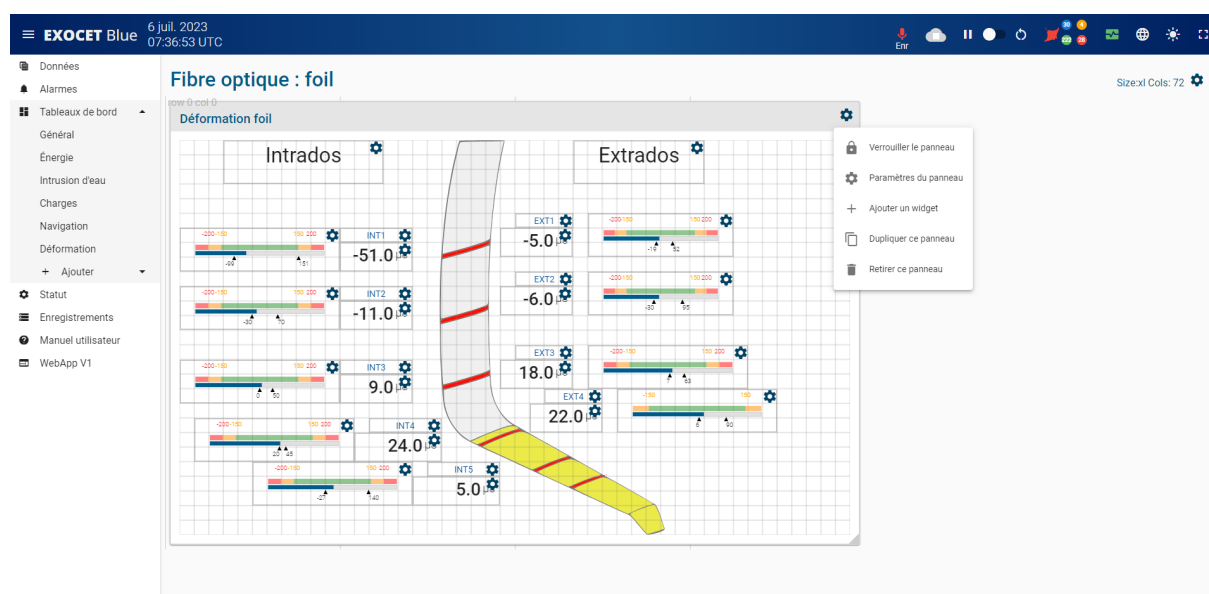
En cliquant sur la roue crantée en haut à droite de l'écran, le menu d'édition du tableau de bord permet de :

- Déverrouiller le tableau de bord (Permet de pouvoir ajouter ou supprimer des panneaux)
- Accéder aux paramètres du tableau de bord (nom, titre anglais et français, affichage dans le menu)
- Exporter un tableau de bord

## Panneau

Une fois le tableau de bord déverrouillé, une roue crantée est disponible en haut à droite de chaque panneau pour accéder à son menu d'édition comprenant :

- Un verouillage déverouillage du panneau (pour ajouter un widget ou figer la géométrie des widget)
- L'accès aux paramètres du panneau
- La duplication du panneau
- La suppression du panneau



**FIG. 52 :** Modification des panneaux

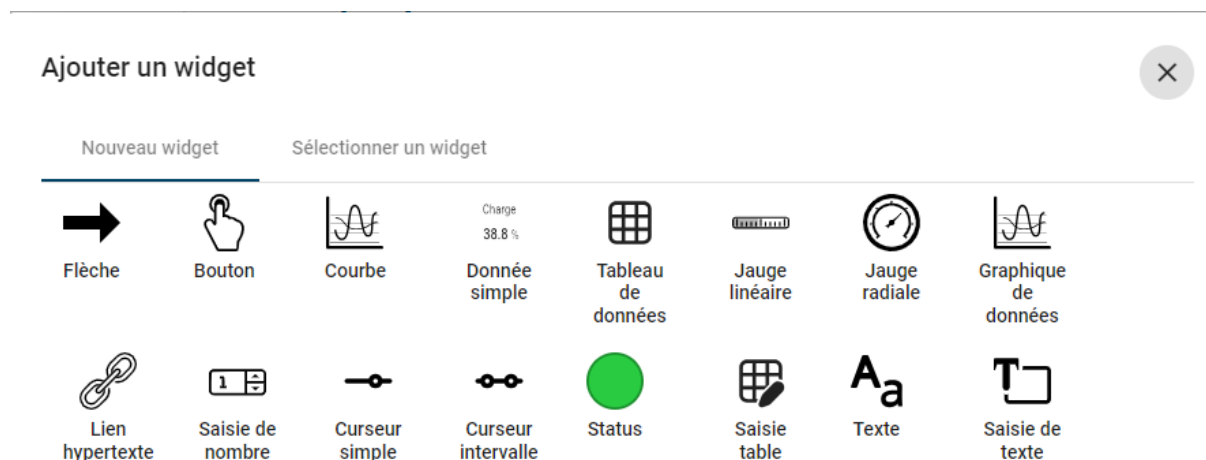
Parmi les paramètres du panneau on trouve :

- Nom et Titre
- Nom système
- Titre anglais
- Titre français
- Affichage
- Mode d'affichage (Grille, libre)
- Couches (réservé pour un usage futur)
- Taille des widget (modifie la taille par défaut des widgets de de panneau)
- Arrière-plan
- Couleur d'arrière plan



- Image d'arrière plan (Il faut vider le cache du navigateur et rafraichir la page si une image du même nom que la précédente est sélectionnée)
- Taille d'affichage (Origine, Ajuster, Remplir, Etirer)
- Répétition d'image (Non, Oui, Horizontale, Verticale)
- Position horizontale (Gauche, Centre, Droite)
- Position verticale (Haut, Centre, Bas)
- Icône. Permet de mettre une petite image à gauche du titre du panneau
- Icône
- Masquer l'icône
- Couleur de l'icône
- Status. Ajoute un indicateur coloré pour indiquer un status
- Gris. Donnée de type *Bitfield* à 0
- Vert. Donnée de type *Bitfield* à 1 ou de tupe *Booléen* à True
- Orange. Donnée de type *Bitfield* à 3
- Rouge. Donnée de type *Bitfield* à 7 ou de tupe *Booléen* à False
- Affichage adaptatif. Permet de régler l'agencement du panneau

## Widget



**FIG. 53 :** bibliothèque de widgets

En cliquant sur le bouton « Ajouter un widget » d'un panel, la bibliothèque de widgets apparaît.

Certains widgets, comme *Donnée simple* ou *Gauge*, sont utilisés pour afficher des données de boîte Manta. D'autres, comme *Button* ou *Slider*, sont utilisés pour contrôler des données de boîte Manta

grâce aux boîtes *Input\_User* (non disponibles dans tous les Exocet).

Cliquer sur le bouton cadenas en haut à droite d'un widget permet d'accéder au menu d'édition du widget pour :

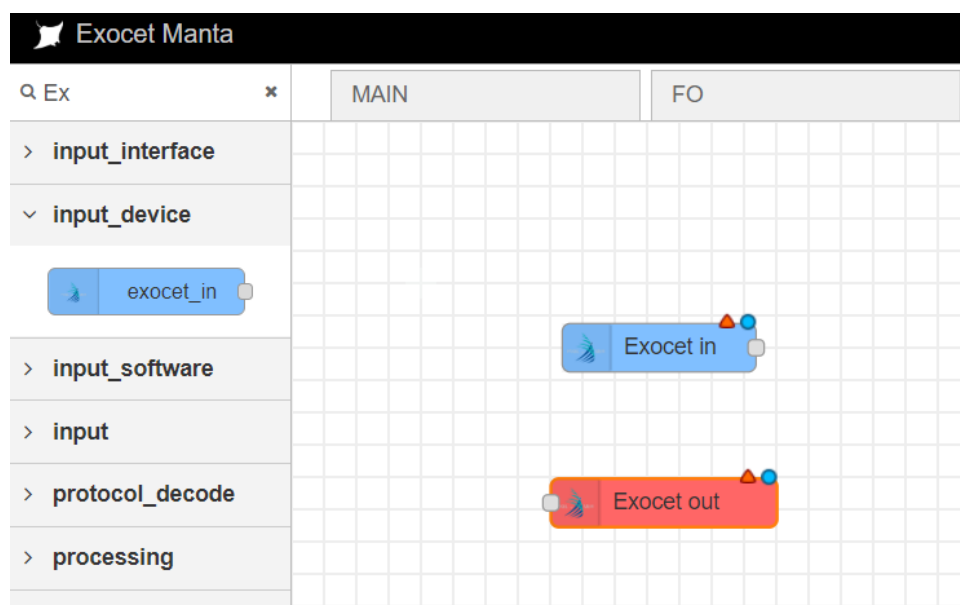
- Configurer les paramètres du widget
- Dupliquer le widget
- Supprimer le widget

Les paramètres de chaque widget comprennent forcément les trois onglets suivants avec des informations à remplir ou sélectionner :

- Nom et Titre
- Nom système
- Titre anglais
- Titre français
- Taille et couleur
- Taille
- Couleur
- Couleur d'arrière-plan
- Affichage du titre
- Masquer le titre
- Utiliser le nom de la donnée comme titre
- Position (Avant, Au-dessus)
- Alignement (Par défaut, Gauche, Centre, Droite)
- Taille du texte (Plus petit, Inchangée, Plus grand)
- Caractère gras
- Largeur du titre en %

D'autres onglets sont disponibles en fonction du widget choisi. Ils permettent de choisir par exemple la donnée à afficher, choisir un visuel de jauge, changer l'apparence visuelle d'un tableau ...

## 8 Communication avec des logiciels tiers



**FIG. 54 :** boîtes Exocet « in » et « out »

Les boîtes Manta « Exocet in » et « Exocet out » ont été conçues pour propager des données entre les produits Exocet, principalement pour envoyer des données de l'Exocet Silver ou Gold à l'Exocet blue afin d'enregistrer ces données.

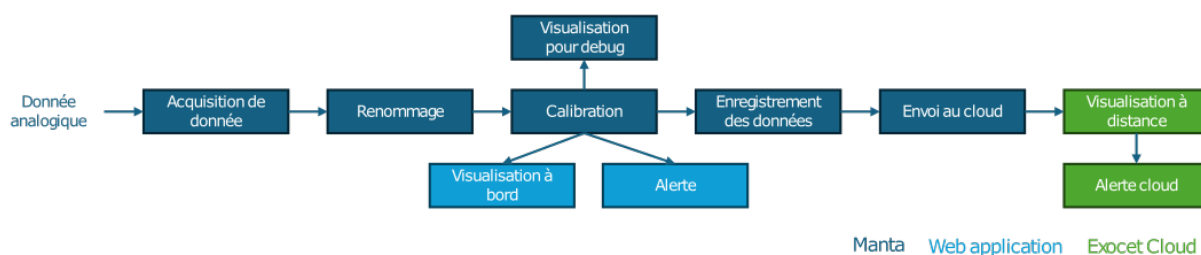
Ces boîtes sont également le moyen le plus simple de communiquer avec des logiciels ou des équipements tiers. La boîte « Exocet out » envoie les données de l'Exocet au logiciel ou à l'équipement tiers. La boîte « Exocet in » reçoit les données d'un logiciel ou d'un équipement tiers.

Les données sont envoyées par Ethernet UDP, en utilisant un protocole propriétaire de Pixel Sur Mer pour la sérialisation des données. La description du protocole peut être fournie sur demande.

## 9 Tutoriel : du capteur au cloud

L'objectif de cette section est de donner un aperçu de la gestion des données par l'Exocet. Il est recommandé de lire les sections **Manta** et **Application web Exocet** avant d'essayer ce tutoriel.

Voici un schéma du flux de données qui sera exploré dans ce tutoriel.

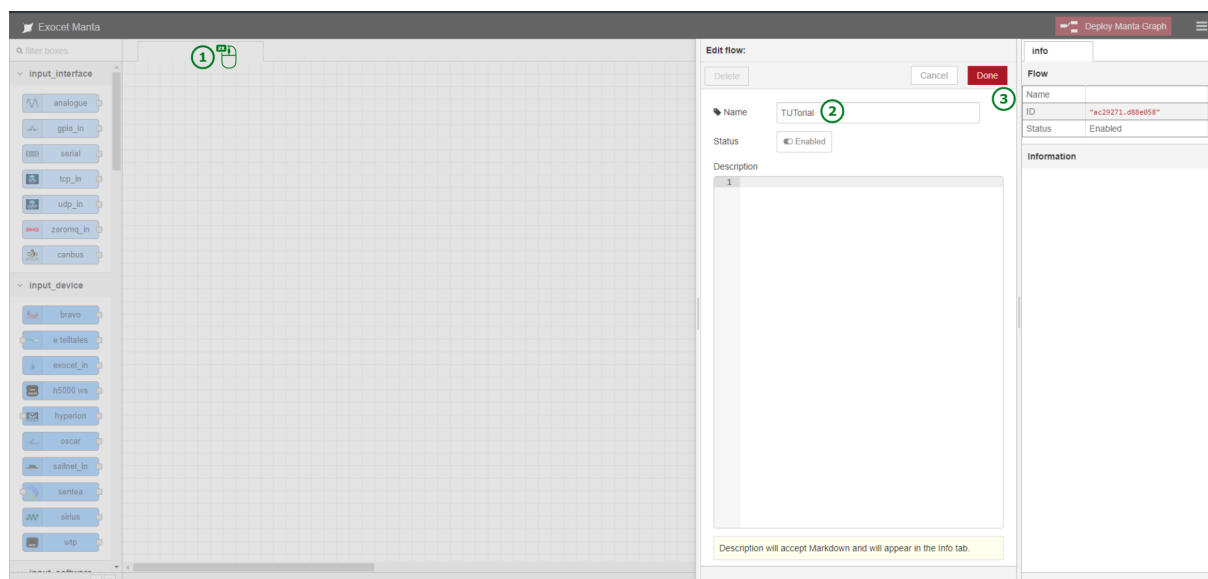


**FIG. 55 :** Flux de données

L'objectif est de lire une valeur analogique, qui décrit un angle de safran, et de l'envoyer au cloud. Les étapes intermédiaires, telles que la calibration, la visualisation embarquée et l'enregistrement des données, seront présentées pour guider l'utilisateur pas à pas à travers les capacités de l'Exocet.

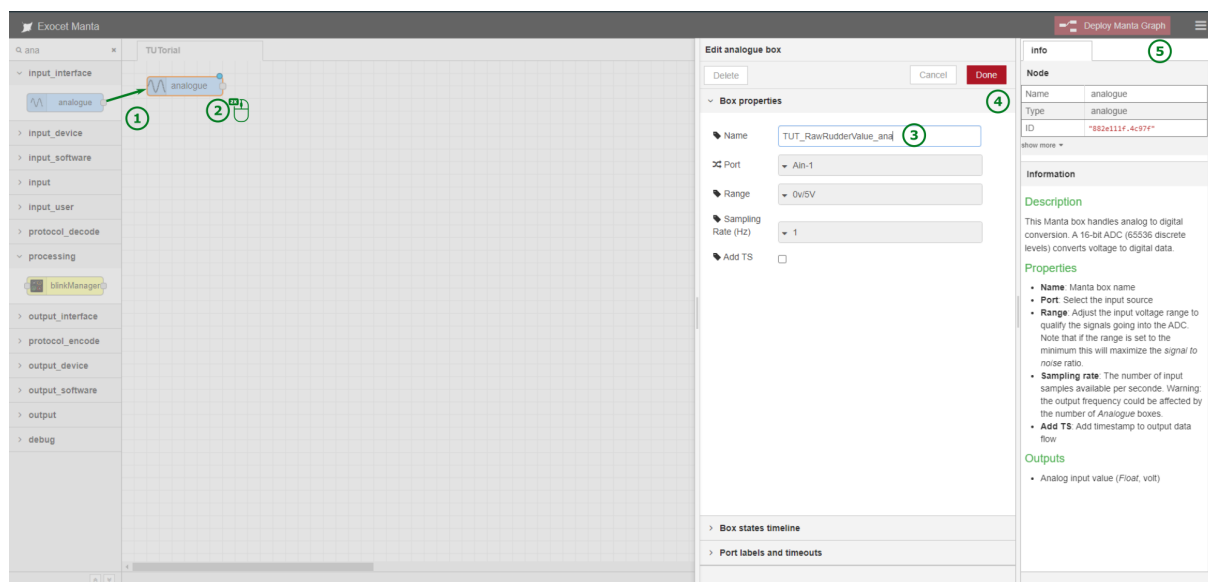
### 9.1 Lire les données analogiques

La première étape consiste à lire les données d'entrée analogiques. Pour ce faire, ouvrez Manta, double-cliquez sur l'en-tête du flux (1), renommez le flux en « TUTORIAL » (2) et validez la modification en appuyant sur le bouton *Done* (3). Tout au long de ce tutoriel, la nomination est basée sur les bonnes pratiques de Manta. Il est fortement recommandé de renommer chaque flux et boîte pour garder le système organisé.



**FIG. 56 :** Renommer le flux Manta

Maintenant, glissez-déposez une boîte analogique (1), double-cliquez dessus (2) et renommez-la en « TUT\_RawRudderValue\_ana » (3). Vous pouvez valider ce changement en appuyant sur *Done* (4) et déployer en appuyant sur le bouton dédié (5).

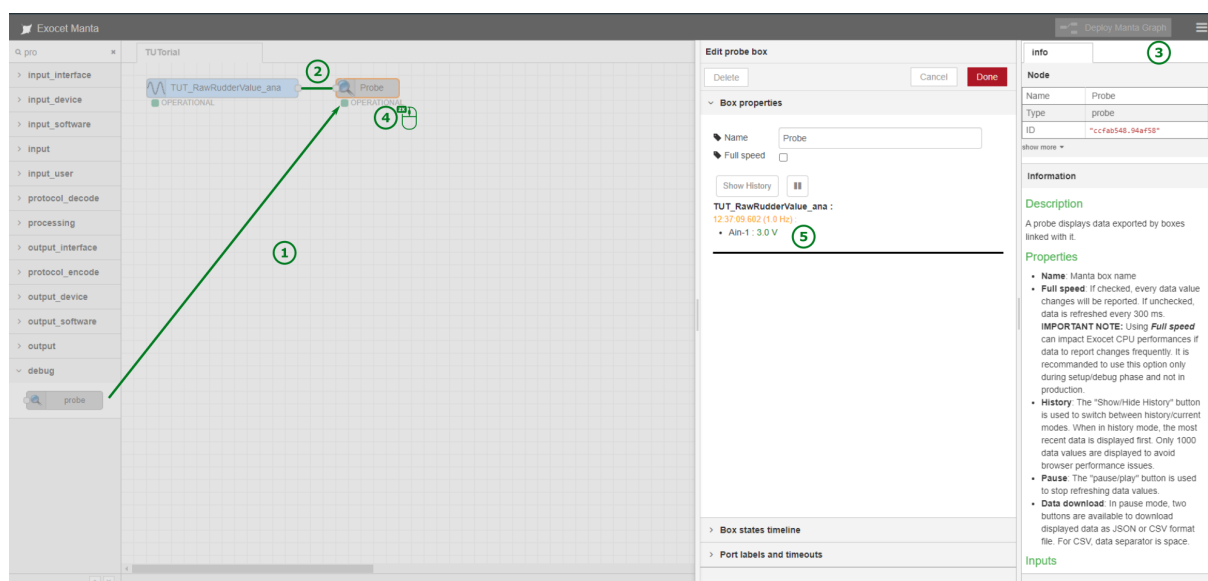


**FIG. 57 :** Lire une valeur analogique

Vous avez maintenant une configuration qui peut lire une valeur analogique entre zéro et cinq volts à un hertz.

## 9.2 Visualiser les données dans une probe

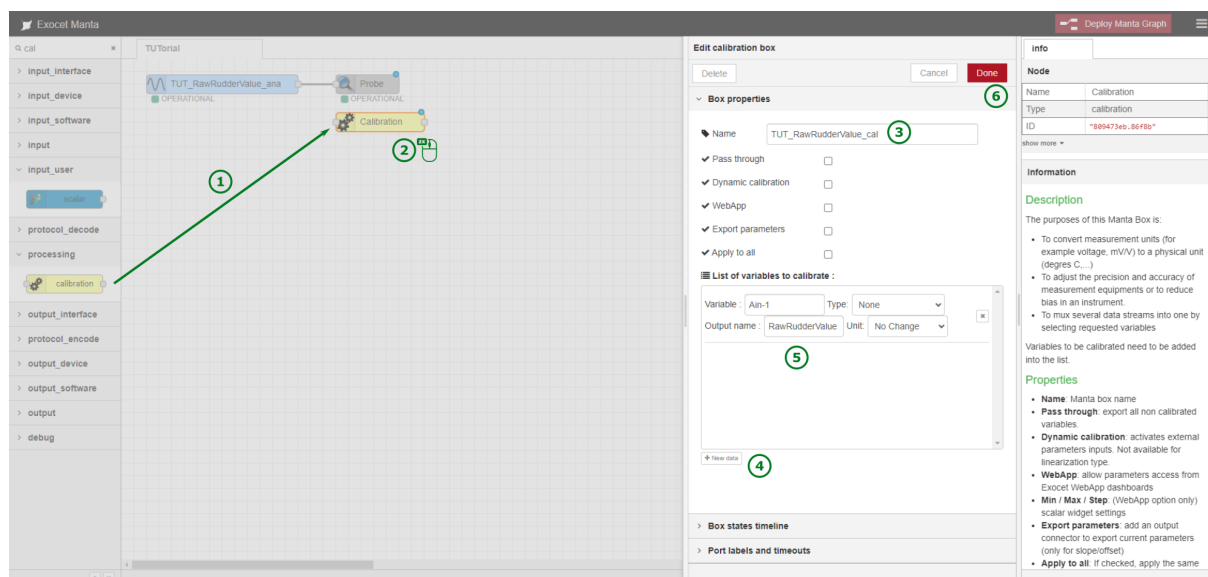
Maintenant que le système acquiert des données, il est temps de les visualiser. Pour visualiser les données d'entrée, glissez-déposez une boîte probe (1) et reliez la sortie *TUT\_RawRudderValue\_ana* à l'entrée *Probe* (2). Pour appliquer la configuration créée, cliquez sur *Deploy Manta Graph* (3). Sous chaque boîte, un statut apparaît, confirmant que ces boîtes fonctionnent désormais. Ouvrez la boîte probe en double-cliquant dessus (4). Les données peuvent être visualisées dans la zone d'édition (5). La boîte *TUT\_RawRudderValue\_ana* publie bien à un hertz une donnée nommée *Ain-1*, image de la tension analogique1.



**FIG. 58 :** Visualisation sonde

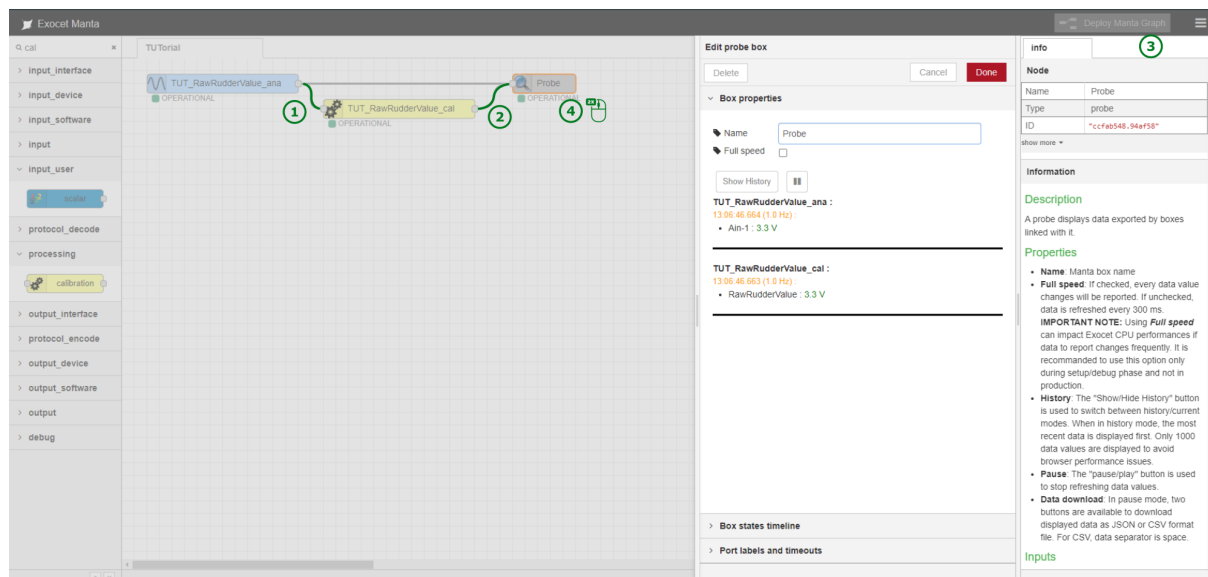
## 9.3 Renommer les données d'entrée

Comme les données publiées par *TUT\_RawRudderValue\_ana* sont nommées « Ain-1 », l'objectif est de les renommer avec un nom pertinent. Glissez-déposez une boîte de Calibration (1), ouvrez-la (2) et renommez-la en *TUT\_RawRudderValue\_cal* (3). Ensuite, cliquez sur *New data* (4). Dans le champ *Variable*, mettez le nom de la donnée d'entrée, appelée « Ain-1 » dans ce contexte. Dans ce tutoriel, la valeur analogique est utilisée pour mesurer un angle de safran, renommez-la en écrivant « RawRudderValue » dans le champ *Output name* (5) et validez en cliquant sur *Done* (6).



**FIG. 59 :** Renommer l'entrée

Vous pouvez maintenant câbler la boîte *TUT\_RawRudderValue\_ana* avec la *TUT\_RawRudderValue\_cal* (1) ainsi que la *TUT\_RawRudderValue\_cal* avec la *Probe* (2). Déployez ce schéma (3) et ouvrez la *Probe* (4). Vous pouvez voir à la fois les données brutes et les données renommées.

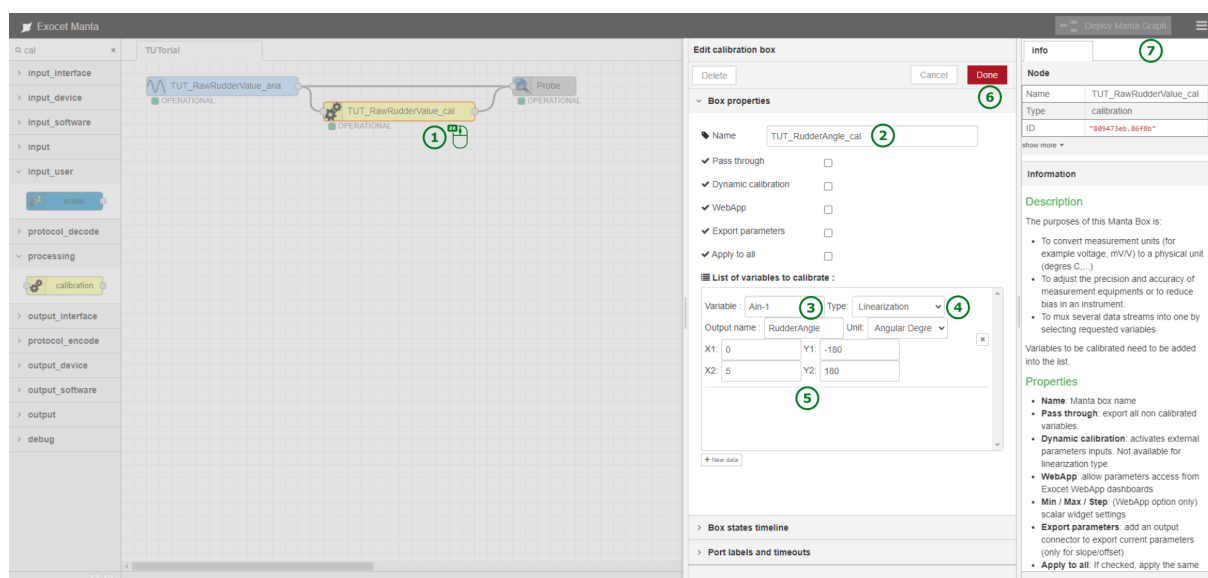


**FIG. 60 :** Visualiser le nouveau nom de donnée

La première étape est terminée ! Jouons maintenant avec ces données.

## 9.4 Calibrer l'entrée

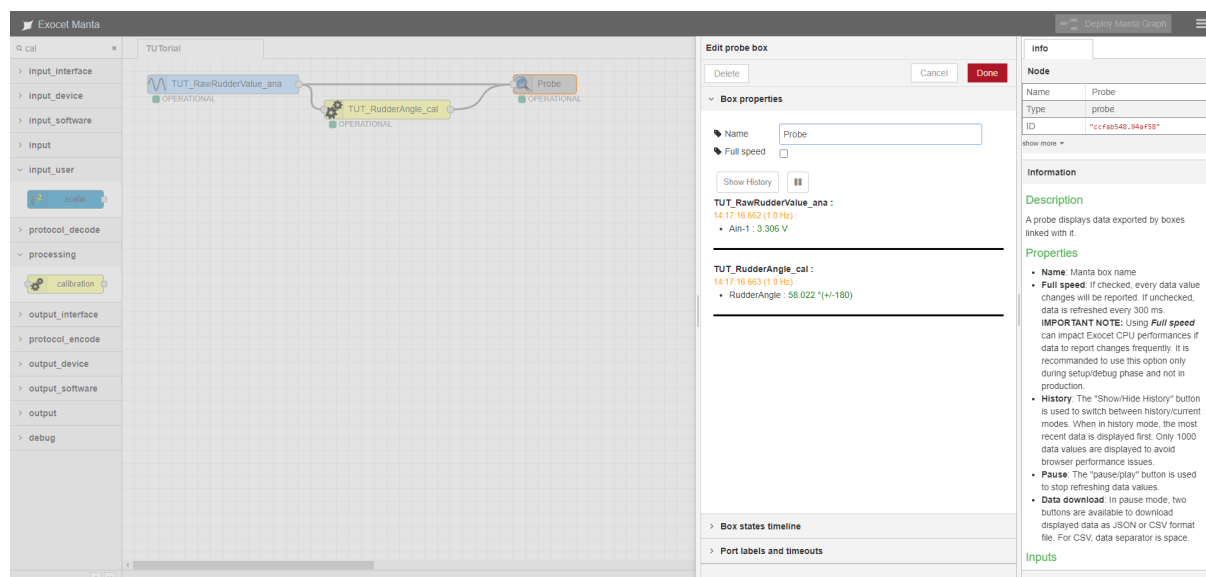
Comme vous l'avez peut-être remarqué, une boîte de Calibration a été précédemment utilisée pour renommer une donnée. Le type de calibration a été laissé à None pour ne pas appliquer de correction, seulement un renommage. Cette fois, les données seront également modifiées. Rouvrez la boîte *TUT\_RawRudderValue\_cal* (1), changez son nom en *TUT\_RudderAngle\_cal* (2) et le nom de sortie en *RudderAngle* (3). Pour la partie calibration, réglez le type de calibration sur *Linearisation* et l'Unité sur *ANGULAR\_DEGREE\_180* (4). En supposant que 0v correspond à  $-180^\circ$  et 5V à  $180^\circ$ , appliquez ces valeurs dans le tableau X1/X2/Y1/Y2 (5). Validez ensuite les changements en appuyant sur le bouton *Done* (6) et déployez-les en appuyant sur le bouton *Deploy Manta Graph* (7).



**FIG. 61 :** Calibration des données d'entrée

Pour une meilleure précision des données affichées dans Manta, rendez-vous sur l'interface graphique de l'Exocet, sur la page Statut et réglez *Décimales par défaut à l'affichage* : à 3. Puis revenez à Manta et ouvrez la sonde. Vous verrez à la fois les données brutes et calibrées avec trois décimales.

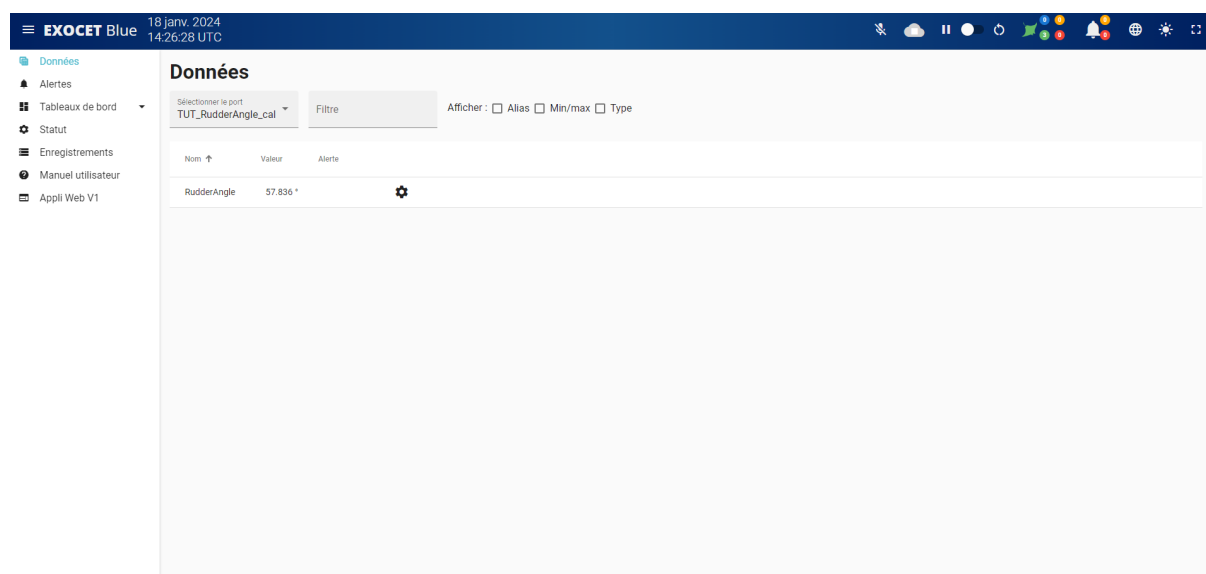




**FIG. 62 :** Visualisation des données calibrées

## 9.5 Visualiser les données dans le tableau de données

Maintenant que la partie logique de notre petit code est faite, concentrerons-nous sur les outils de visualisation offerts par l'Exocet. Allez sur l'application web à la page Données. Dans le champ *Sélectionner le port*, indiquez le nom de la sortie d'une boîte que vous souhaitez surveiller (« TUT\_RudderAngle\_cal » pour cet exemple). La liste des données exportées par le connecteur de sortie sélectionné est maintenant accessible, permettant d'afficher la valeur actuelle et d'accéder à la configuration des données. Notez que si une boîte a un seul connecteur de sortie, il porte le même nom que la boîte, sinon les connecteurs ont un suffixe #NUMBER.

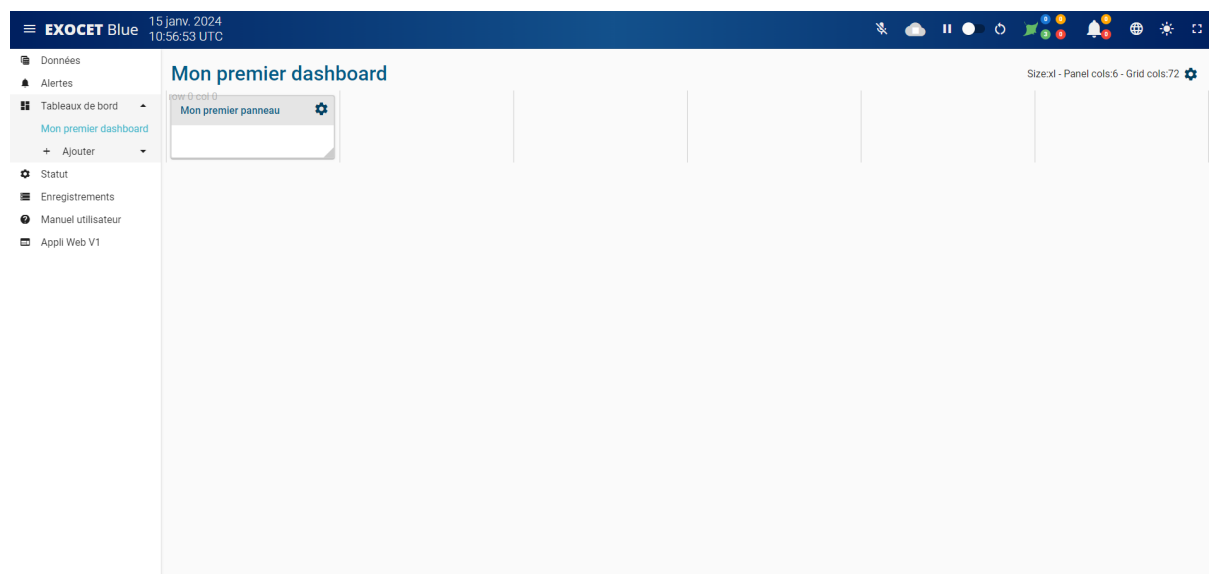


**FIG. 63 :** Visualisation du tableau de données

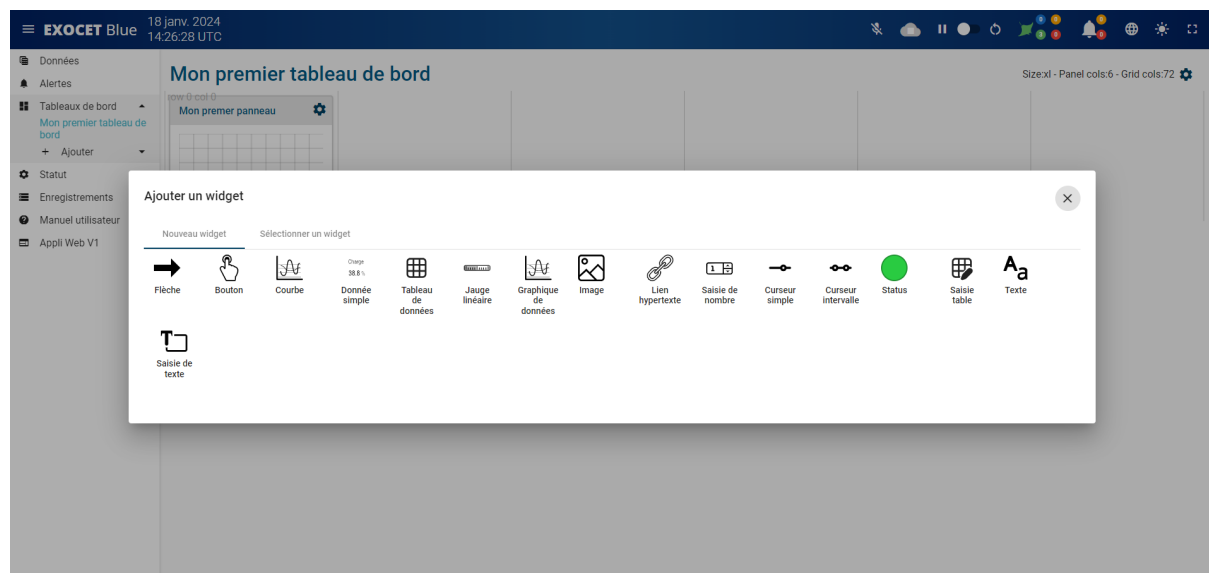
## 9.6 Visualiser les données dans un tableau de bord

La valeur analogique est maintenant convertie en sa représentation physique réelle. Il est temps de l'afficher sur l'interface graphique ! Vous devez d'abord créer un tableau de bord. Sur l'application web, cliquez sur Tableaux de bord/Ajouter/Nouveau. Dans le menu qui s'ouvre, remplissez le champ *Nom système* avec « my-first-dashboard » et le titre par « Mon premier tableau de bord ». Cliquez sur *Enregistrer*. Superbe, vous avez votre premier tableau de bord personnalisé.

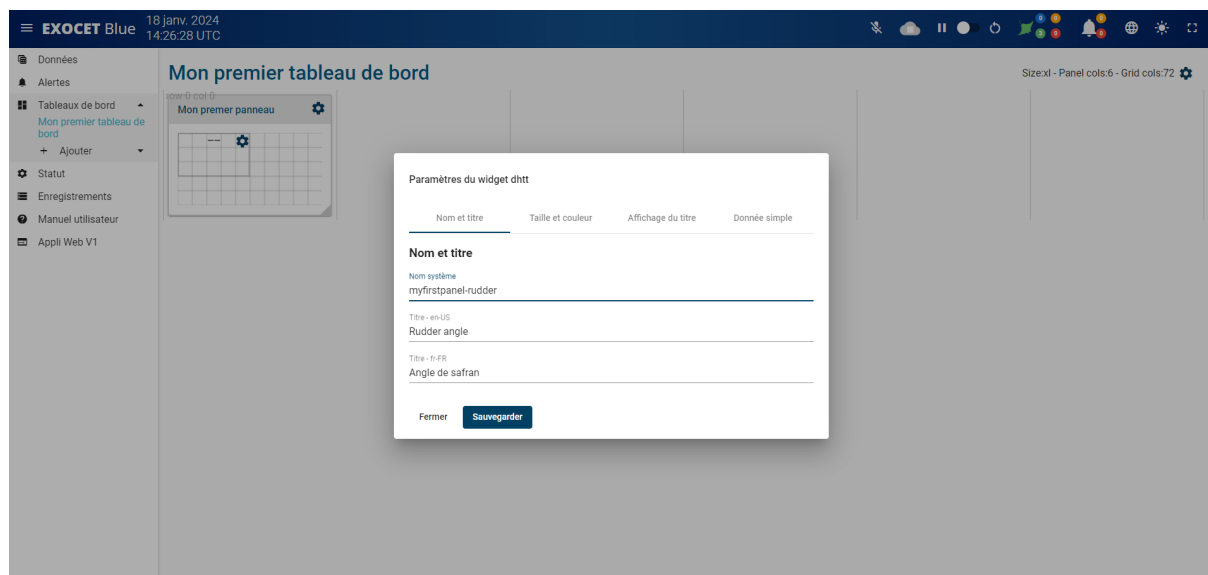
En haut à droite de la zone du tableau de bord, cliquez sur la roue dentée et déverrouillez le tableau de bord. Le tableau de bord est maintenant modifiable. Cliquez à nouveau sur la roue dentée et choisissez l'option *Ajouter un panneau*. Comme précédemment, dans le menu qui s'ouvre, remplissez le champ *Nom système* avec « my-first-panel » et le titre par « Mon premier panneau ». Cliquez sur *Enregistrer*. Les panneaux et tableaux de bord permettent de trouver la disposition souhaitée qui répond à vos besoins.

**FIG. 64 :** Création d'un panneau

La zone modifiable d'un panneau est composée d'une grille. Le coin inférieur droit d'un panneau peut être cliqué et tiré pour redimensionner le panneau. Essayez de le faire pour obtenir une grille de 10x5. Comme nous avons maintenant plus d'espace d'affichage, cliquez sur la roue dentée du panneau et sélectionnez *Ajouter un widget*. Plusieurs widgets sont disponibles. Comme nous souhaitons afficher une seule donnée numérique, sélectionnez *Donnée simple*.

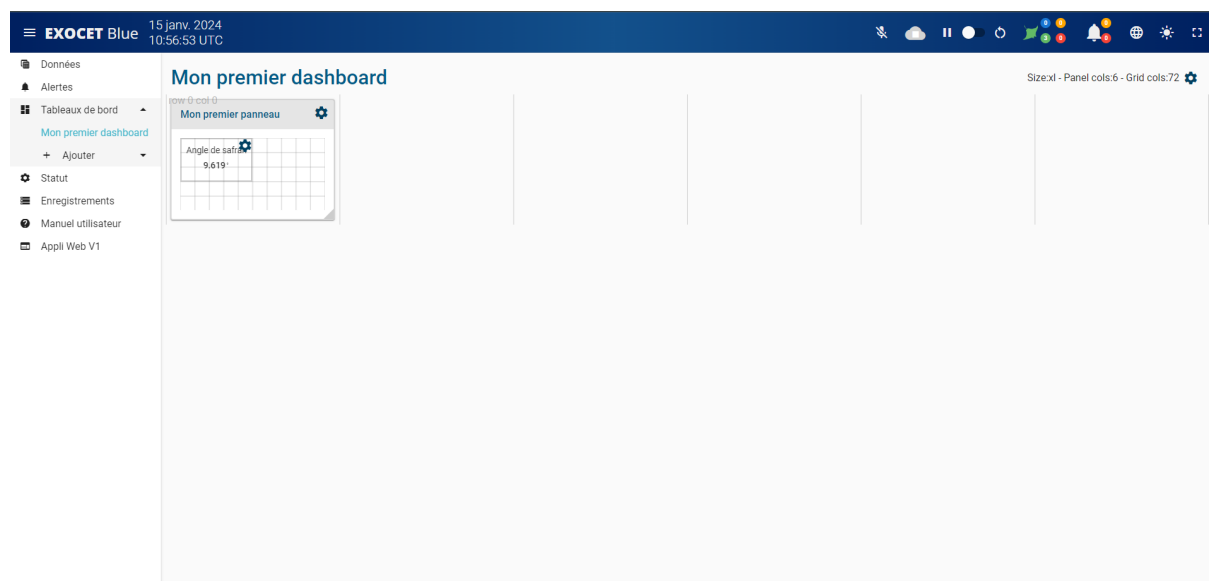
**FIG. 65 :** Widgets disponibles

Le widget de donnée simple apparaît avec une petite dimension. Redimensionnez-le pour correspondre à 5x3 carrés dans la grille. Pour afficher l'angle du safran sur le widget, cliquez sur la roue dentée du widget et cliquez sur *Paramètres du widget*. Dans la section *Nom et titre*, remplissez le nom système avec « myfirstpanel-rudder » et le titre avec « Angle du safran ».



**FIG. 66 :** Définir le titre du widget

Dans la section *Donnée simple*, sélectionnez la boîte qui exporte la donnée souhaitée (*TUT\_RudderAngle\_cal*). Comme « RudderAngle » est la seule donnée exportée par cette boîte, elle est automatiquement choisie. Vous pouvez maintenant enregistrer ces modifications.

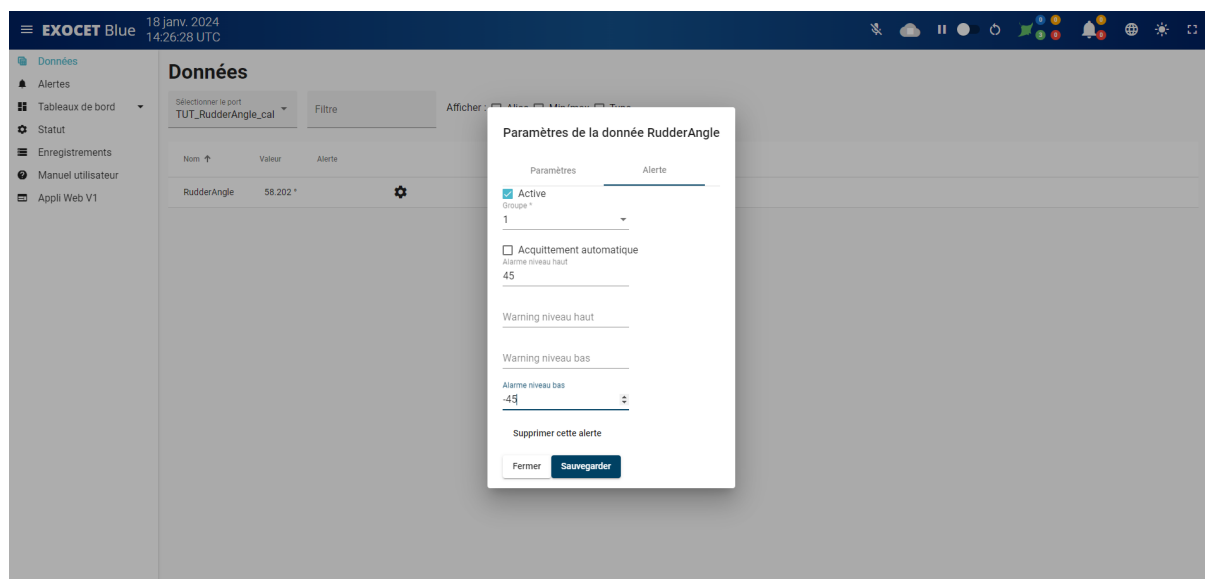


**FIG. 67 :** Visualisation des données du widget

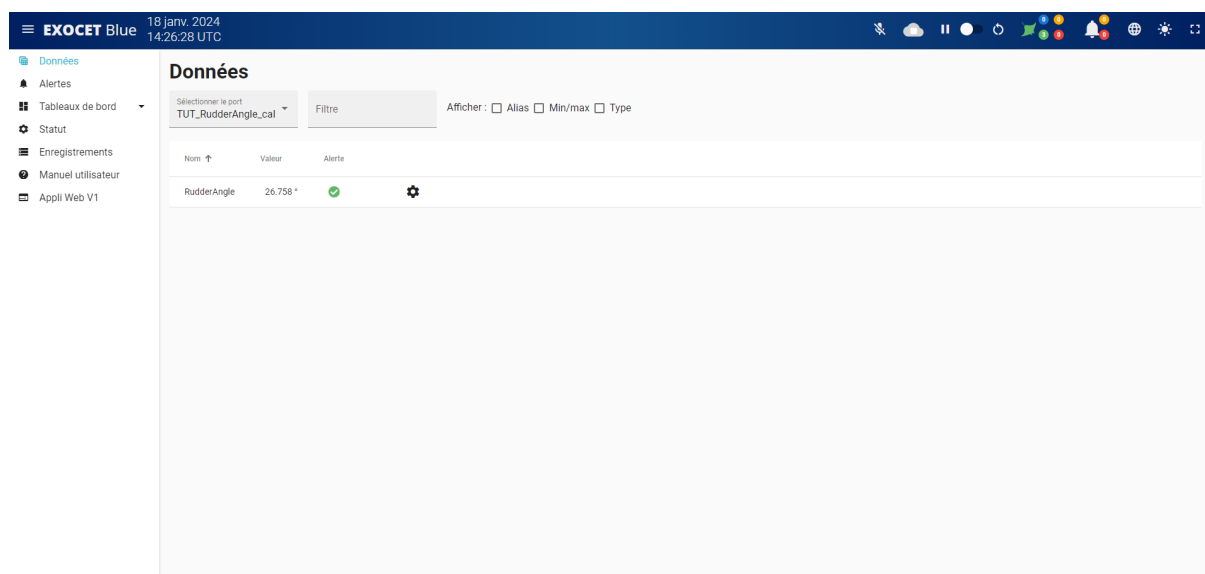
Félicitations, les données de l'angle du safran sont maintenant affichées ! Pour fermer le mode d'édition, cliquez sur la roue dentée du tableau de bord et verrouillez le tableau de bord. N'hésitez pas à jouer avec l'éditeur plus tard.

## 9.7 Configuration d'une alerte

Imaginez que nous souhaitions placer une alerte sur les données du safran. Pour cet exemple, nous pouvons considérer que l'angle réel du safran est compris entre  $-45^\circ$  et  $45^\circ$ . Si nous voulons être avertis que l'entrée du capteur est incorrecte, nous allons définir une alerte au niveau d'alarme si les données de l'angle du safran sont hors de cette plage. Pour ce faire, rendez-vous sur la page de données de l'application web et affichez RudderAngle comme décrit dans les deux sous-sections précédentes. Ensuite, accédez aux paramètres des données (roue dentée noire sur la même ligne). Dans la section Alertes, cliquez sur « Créer une alerte ». Pour cet exemple, attribuez l'alerte au groupe 1 et réglez le *Seuil d'alarme haut* à 45 et le *Seuil d'alarme bas* à -45, puis enregistrez les modifications en cliquant sur le bouton *Sauvegarder*.

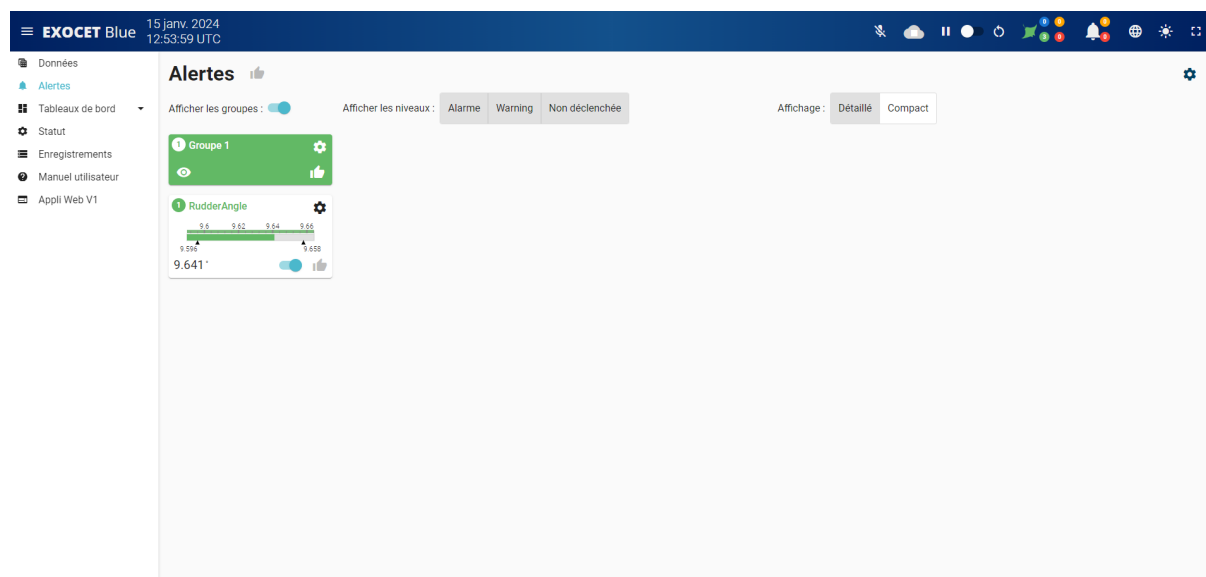
**FIG. 68 :** Configurer une alerte

L'alarme est maintenant créée, un statut apparaît pour montrer l'état actuel.

**FIG. 69 :** Vérification de l'alerte

## 9.8 Visualiser l'alerte

Visualisons la nouvelle alerte sur la page dédiée. Allez à la page des alertes et vous pouvez voir votre alerte affichée en vert ou en rouge selon l'état de la valeur de RudderAngle.



**FIG. 70 :** Visualisation de l'alerte dans l'application web

## 9.9 Enregistrement des données

Pour enregistrer les données souhaitées sur la carte SD de l'Exocet, nous devons ajouter une nouvelle boîte dans le flux Manta. Passez à la page Manta et ajoutez une boîte *datalogger* (1). Reliez la boîte *TUT\_RudderAngle\_cal* au premier connecteur de la boîte datalogger (2) et ouvrez la boîte datalogger (3). Renommez la boîte en « *TUT\_DataLogger\_dtl* » (4). Pour le moment, nous conserverons les paramètres par défaut de la boîte. Lisez la description à droite pour avoir plus d'explications sur chaque paramètre. Vous pouvez valider les paramètres de la boîte en cliquant sur le bouton *Terminé* (5) et déployer (6). L'angle du safran est maintenant enregistré sur l'Exocet.

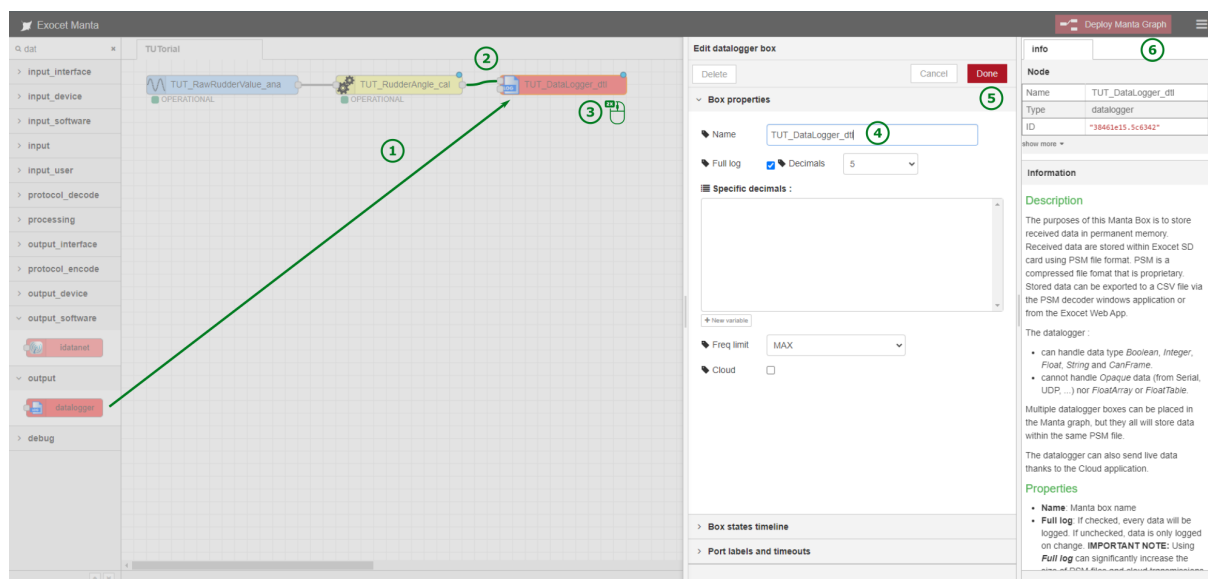


FIG. 71 : Configuration du datalogger

## 9.10 Envoyer des données au cloud

Pour envoyer des données au cloud, vous avez d'abord besoin d'un compte cloud. Une fois ce compte activé, l'Exocet peut exporter des données vers le cloud tout en les sauvegardant sur la carte SD. Dans le flux Manta, ouvrez la boîte *TUT\_DataLogger\_dtl* (1) et cochez la case nommée *Cloud* (2). Créez une nouvelle configuration cloud en cliquant sur le stylo à côté de *Cloud conf* (3).

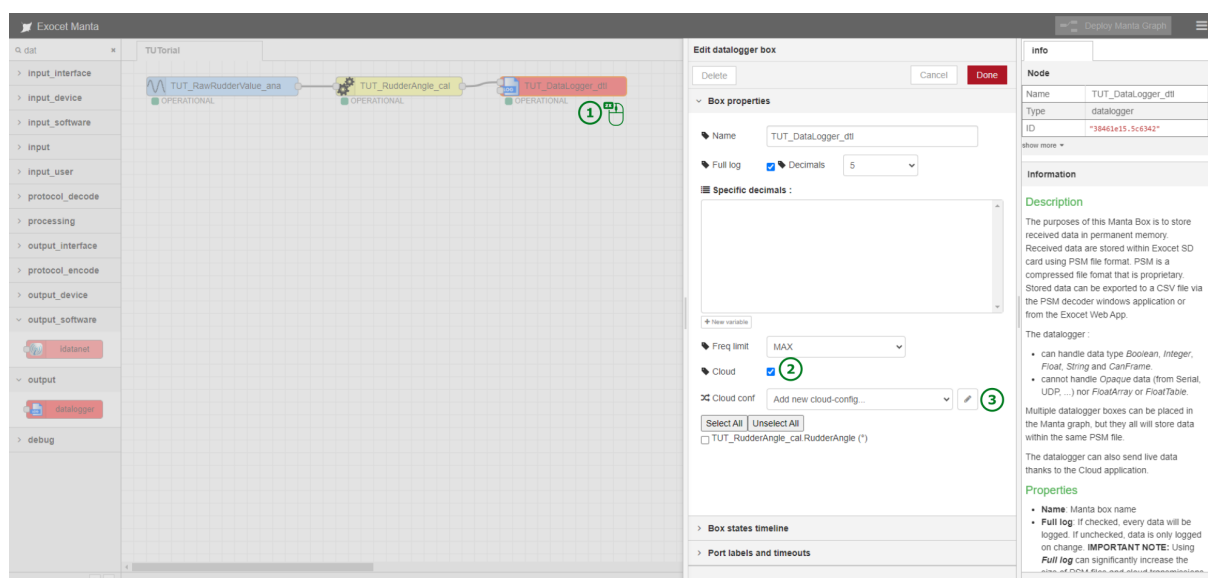
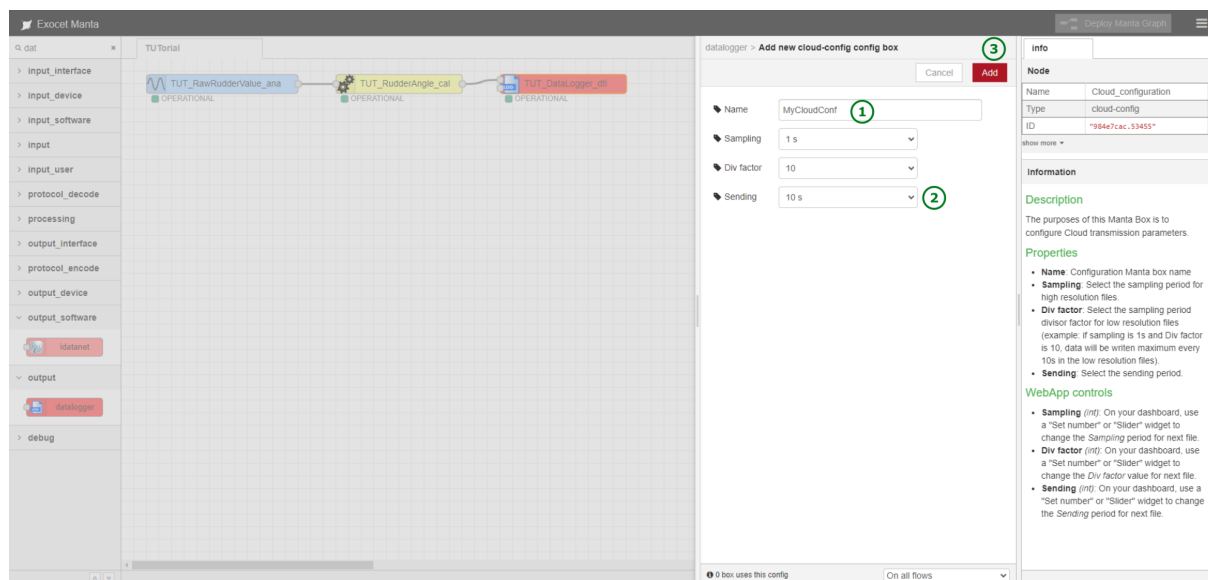


FIG. 72 : Ajouter une configuration cloud

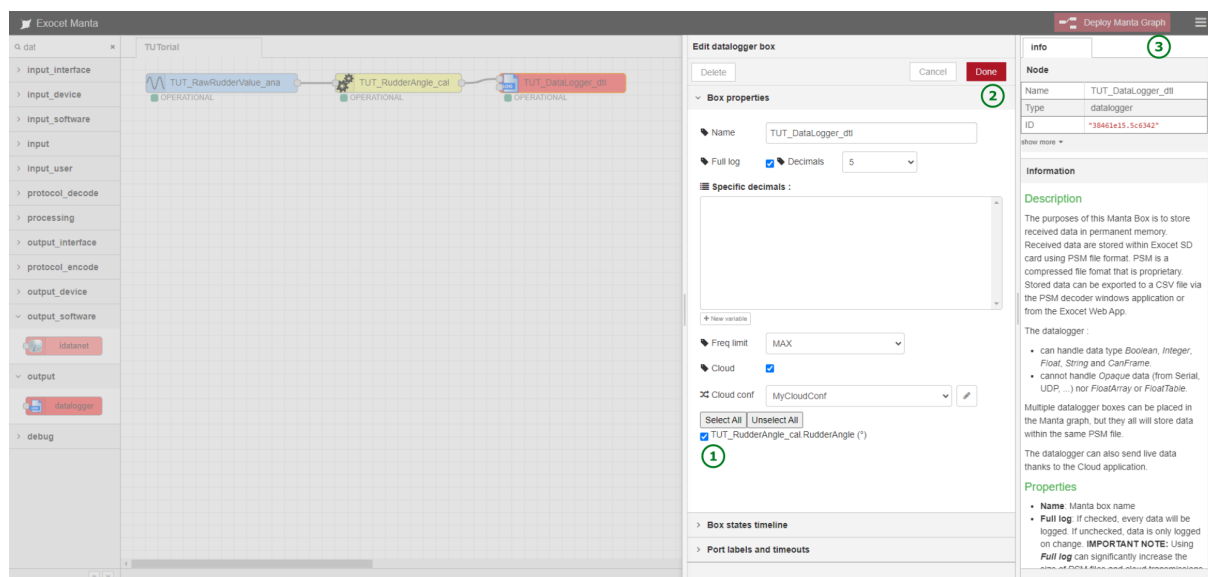


Dans le menu de configuration cloud, changez le nom en « MyCloudConf » (1). Pour cet exemple (non recommandé sur les bateaux), changez le paramètre *Sending* à 10s (2). Soyez prudent : plus vous envoyez de données au cloud, plus vous utiliserez de données Internet. Cliquez sur *Add* pour ajouter la nouvelle configuration (3).



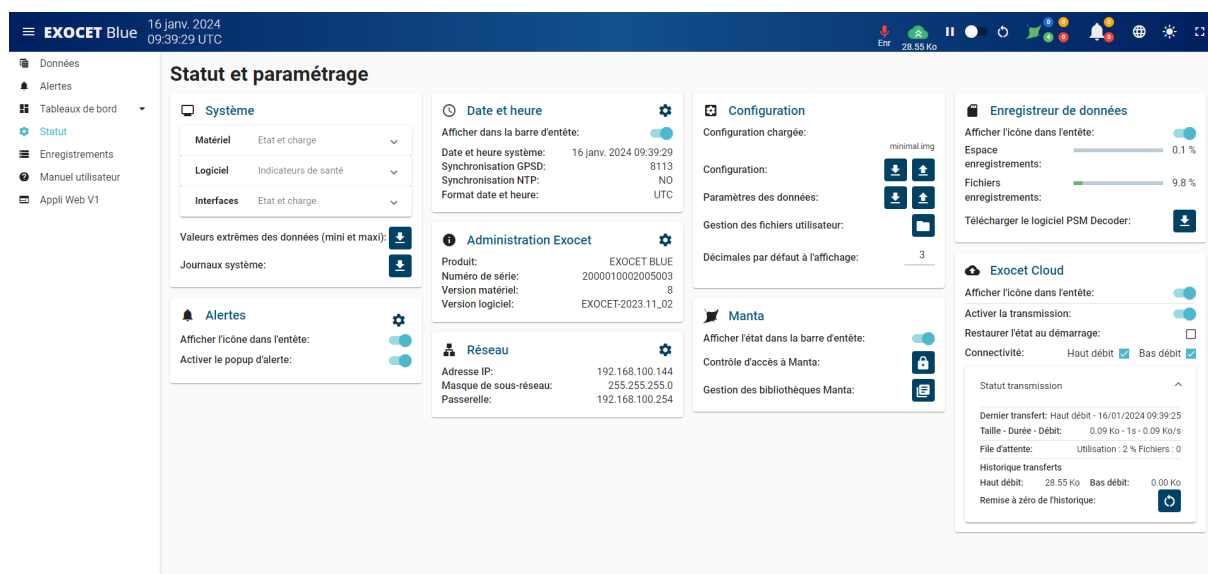
**FIG. 73 :** Modifier la configuration cloud

Toutes les données disponibles dans la boîte *dataLogger* sont prêtes à être envoyées au cloud. Sélectionnez *TUT\_RudderAngle\_cal.RudderAngle (°)* (1), validez ces paramètres en cliquant sur le bouton *Done* (2) et déployez les changements (3).



**FIG. 74 :** Configuration cloud valide

Pour activer le transfert de données au cloud, passez à la page Statut de l'interface web et activez *Activer la transmission* dans la section *Exocet Cloud*. Vous devriez voir que l'étiquette *Dernier transfert* a été mise à jour 10 secondes plus tard.

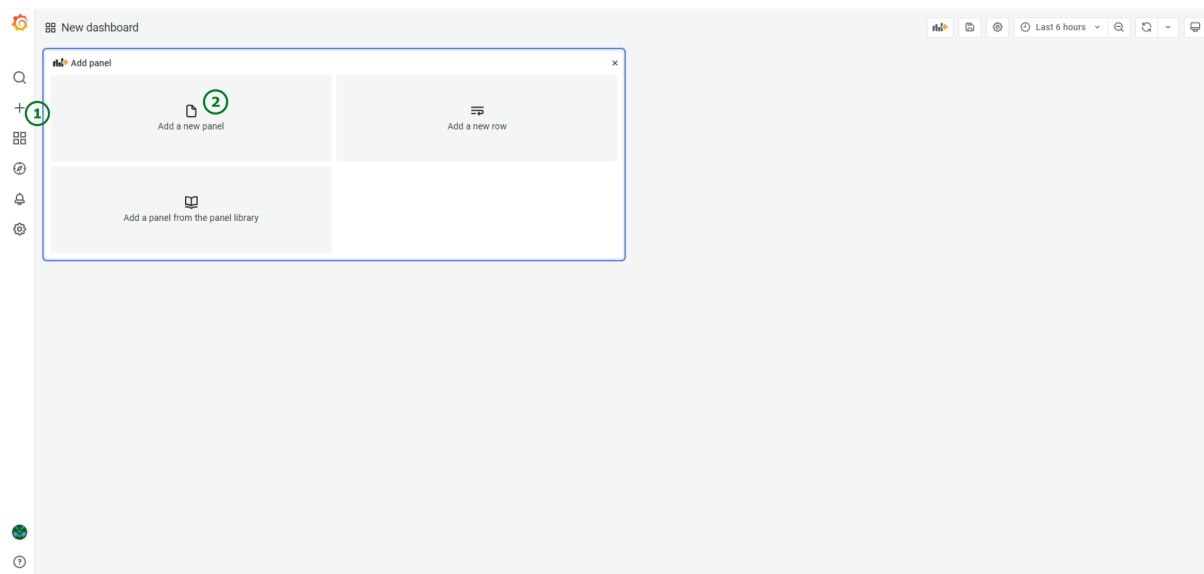


**FIG. 75 :** Connectivité cloud

Pour obtenir plus d'informations sur le cloud Exocet, téléchargez et lisez la documentation dédiée : <https://exocet.cloud/documentation>.

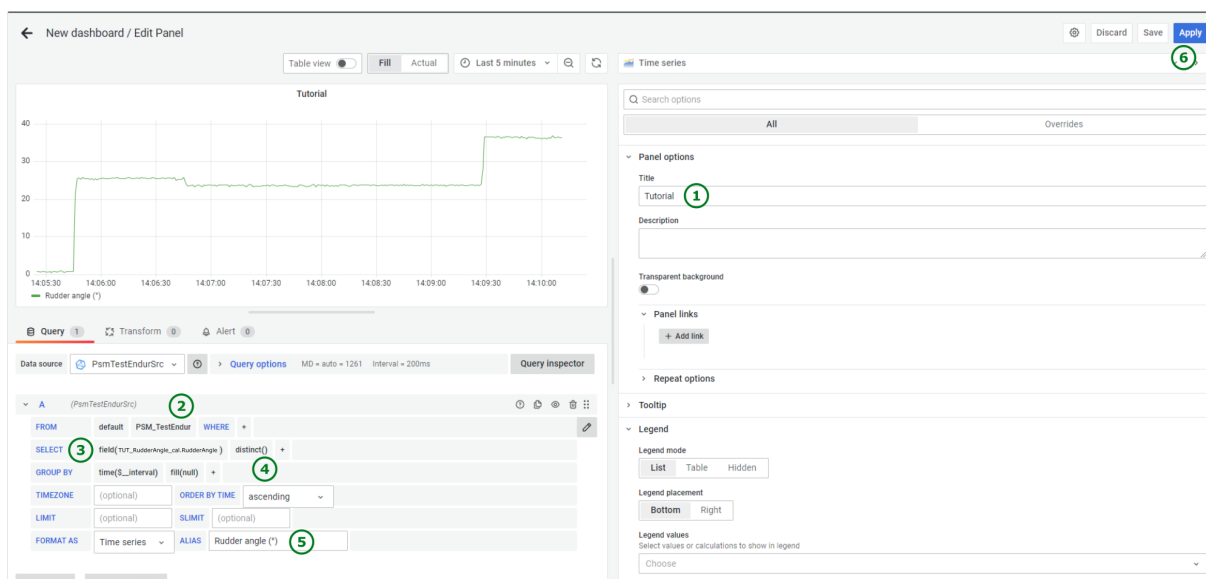
## 9.11 Visualiser les données cloud avec Grafana

Connectez-vous à votre organisation sur <https://exocet.cloud>. Créez un tableau de bord et choisissez *Ajouter un panneau*.

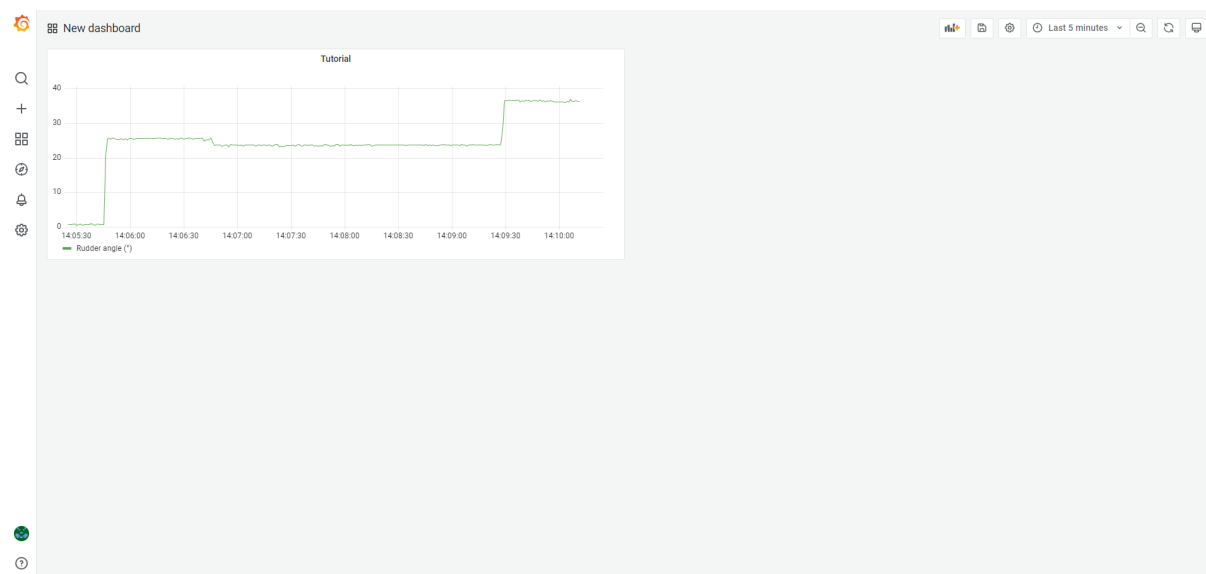


**FIG. 76 :** Ajout d'un panneau Grafana

Vous êtes maintenant sur la vue d'édition. Sur le côté droit, tous les paramètres se réfèrent à l'apparence du panneau. En haut à gauche, un aperçu du panneau est affiché. En bas à gauche sont listées les requêtes pour sélectionner les variables à afficher. Commencez par définir le titre sur « Tutoriel » (1). Ensuite, allez dans la mise en page de la requête et sélectionnez votre base de données en cliquant sur *select measurements* (2). Le nom diffèrera de cet exemple. L'étape suivante consiste à sélectionner les données du safran. Cliquez sur *field(value)* et sélectionnez *TUT\_RudderAngle\_cal.RudderAngle* (3). Pour afficher toutes les données plutôt qu'une moyenne locale, cliquez sur le + à côté de *mean()* pour choisir *distinct()* (4). Enfin, remplissez le champ *Alias* avec « Angle du safran (°) » (5). Vérifiez dans l'aperçu que « Angle du safran (°) » n'est pas désélectionné. Cliquez ensuite sur *Apply* en haut à droite (6).

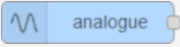



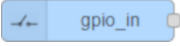



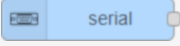







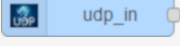



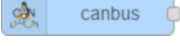



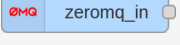



**FIG. 77 :** Édition du panneau Grafana

Vous avez maintenant toute la chaîne de données en fonctionnement !

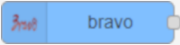







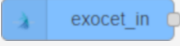



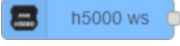



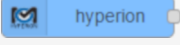

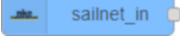



**FIG. 78 :** Visualisation sur le Cloud



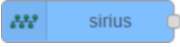





## 10 Manta boxes overview table

### 10.1 Input interfaces

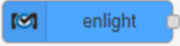

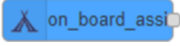



Box	Description	Exocet
 analogue	Convert an analog signal to a digital value	  
 gpio_in	Manage digital input	  
 serial	Manage RS232 or RS422 serial input	  
 tcp_in	Get data from a TCP interface	  
 udp_in	Get data from an UDP interface	  
 canbus	Get data from an CAN 2.0 bus interface	  
 zeromq_in	Get data from a ZeroMQ interface	  

## 10.2 Input devices

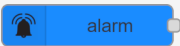











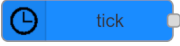



Box	Description	Exocet
 bravo	Import data from Bravo4 sailing instrument processor	  
 e telltales	Export angles for a set of a-telltales	  
 exocet_in	Import data from another Exocet Blue, Silver or Gold	  
 h5000 ws	Import B&G H5000 sailing instrument variables	  
 hyperion	Read microstrains or temperatures from Hyperion, the Micron Optics optical fiber interrogator	
 sailnet_in	Imports NKE data from SailNet protocol over Ethernet	  

Box	Description	Exocet
 sentea	Read microstrains or temperatures from Sentea optical fiber interrogator	
 sirius	Import data from Pixel Sur Mer multiple sensors gateway	
 wtp	Import B&G WTP3 sailing instrument variables	  





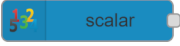



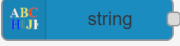







### 10.3 Input softwares

Box	Description	Exocet
 enlight	Read microstrains or temperatures from Enlight, the Micron Optics optical interrogator PC software	
 on_board_assi	Receive events from OnBoardAssistant Sailing performance software.	  

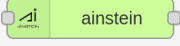




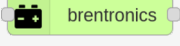

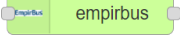

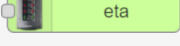

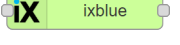











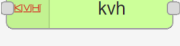



### 10.4 Input

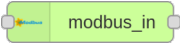



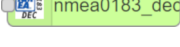

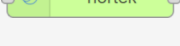



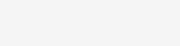



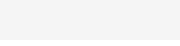







Box	Description	Exocet
 alarm	Inform if at least one alarm or warning is active	  
 link	Virtual wires to import data from another flow	  
 manta_status	Inform if at least one Manta box is on error state or on warning state	  
 tick	Generate a periodic tick	  

### 10.5 Input User

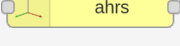

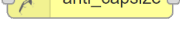

Box	Description	Exocet
 button	Inject a boolean value that can be updated by user from Web App dashboard	  
 scalar	Inject an integer or decimal value that can be updated by user from Web App dashboard	  
 string	Inject a text value that can be updated by user from Web App dashboard	  
 table	Inject a table that can be updated by user from Web App dashboard	  

## 10.6 Protocol Decode



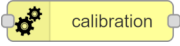

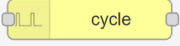

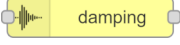

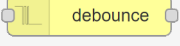

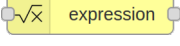

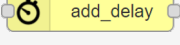

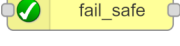

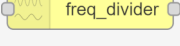

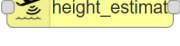

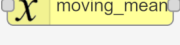

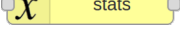

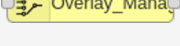

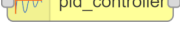

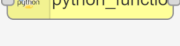

Box	Description	Exocet
 ainstein	Decode Einstein altimeter UART & CAN protocol	
 ais	Decode AIS sentences payload	 
 brentronics	Decode Brentronics battery status sent over CAN bus	
 empirbus	Decode EmpirBus user data sent over CAN bus	
 eta	Decode ETA Power Distribution Systems sent over CAN bus	
 ixblue	Decode several IxBlue Inertial Navigation System protocols	  
	Extract data from JSON	  
 keel	Decode data of IMOCA keel management system	  
 kvh	Decode some ANPP binary messages from KVH Inertial Navigation System.	  



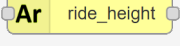





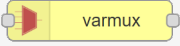





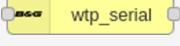



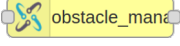

Box	Description	Exocet
 modbus_in	Import data from remote Modbus device	
 net2000_in	Read all variables, parameters and events of a selected Net2000 source	
 nmea0183_dec	Decode NMEA0183 sentences sent over Ethernet/UDP or serial	
 nortek	Decode some Nortek DVL binary messages	
 opc_in	Import data from remote OPC-UA device	
 oscar	Decode some data from Oscar messages	
 sbg	Decode some SBG Inertial Navigation System binary messages	
 senix	Decode Senix ASCII protocol	
 victron	Decode Victron VE.Direct protocol	
 wattandsea	Decode Watt&Sea aero and hydro generator status sent via serial interface	
 williamson	Decode williamson battery status sent over CAN bus	

## 10.7 Processing

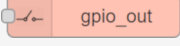



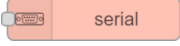







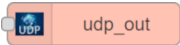



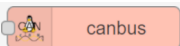



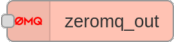



Box	Description	Exocet
 ahrs	Calibrate AHRS position and apply magnetic corrections	
 anti_capsize	Angular piece of anti-capsize system	



Box	Description	Exocet
 blinkManager	Manage Blink Powerkey Pro from NMEA2000	
 calibration	Adjust data precision and convert units	
 cycle	Execute a periodic digital signal when its input goes to True state.	
 damping	Filtering for scalar or complex data	
 debounce	Execute a debouncing filter on specified variables	
 expression	Use C++ Mathematical Expression Toolkit to execute arithmetic operations, functions and processes	
 add_delay	Delays a signal	
 fail_safe	Manage several redundant inputs to ensure a safe output based on port timeouts	
 freq_divider	Perform a sampling frequency division on incoming data	
 height_estimat	compute the height above the water of points of interest	
 moving_mean	Perform statistics on selected variables	
 stats	Perform statistics on data dictionary	
 Overlay_Manage	Manages several pilot overlays	
 pid_controller	PID controller	
 python_function	A function block where you can write Python 2.7 code to do interesting things	

Box	Description	Exocet
 regulation_perp	Pilot overlay plateforme for the implementation of multi-objective regulation performance rule	
 Ar ride_height	compute the height above the water of points of interest	
 varfilter	Extract one or more variables from an input data stream	  
 varmux	Mux several input data streams	  
 Ar vector_man	Rotate a 3D vector by providing 3 rotations angles for each axis	
 wtp_serial	Manage WTP serial module from NMEA2000	  
 obstacle_man	Manage obstacle detection	



















## 10.8 Output Interface

Box	Description	Exocet
 gpio_out	Manage digital output	  
 serial	Send data to the specified RS232 or RS422 port	  
 tcp_out	Send data to a TCP interface	  
 udp_out	Send data to an UDP interface	  
 canbus	Send data to a CAN 2.0 bus interface	  
 zeromq_out	Send data to a ZeroMQ interface	  

Box	Description	Exocet
-----	-------------	--------





















## 10.9 Protocole Encode







Box	Description	Exocet
-----	-------------	--------

 ascii_encoder	Generic ASCII protocol encoder	  
 empirbus_out	Encode EmpirBus user data to send over CAN bus	
 json_enc	Encode data to JSON	  
 modbus_out	Export data to remote Modbus device	  
 net2000_out	Send variable, parameter or event to Net2000	  









## 10.10 Output Device

Box	Description	Exocet
-----	-------------	--------







 bravo	Export data to Bravo 4 Sailing Instrument Processor	  
 display_bandg	Send custom variables to B&G displays	  
 display_garmin	Send custom variables to Garmin displays	  
 display_nke	Send custom variables to NKE displays	  
 exocet_out	Export data to another Exocet Blue, Silver or Gold	  

Box	Description	Exocet
 pilot_overlay_r	Ease of use NKE pilot interface	
 pilot_overlay_b	Ease of use B&G pilot interface	
 pilot_settings	Ease of use B&G pilot interface	





## 10.11 Output Software





Box	Description	Exocet
 adrena_xdr	Export data to Adrena navigation software via XDR protocol	  
 idatanet	Export data to iDataNet iPhone/iPad app via Ethernet/UDP	  

## 10.12 Output



















Box	Description	Exocet
 link	Virtual wires to export data to another flow	  
 datalogger	Store data within Exocet persistent memory	

## 10.13 Debug

Box	Description	Exocet
 comment	A box you can use to add comments to your Manta graph	  

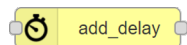
Box	Description	Exocet
 probe	Display data exported by one or several boxes	
 system	Export system usage data	

## 10.14 wind

Box	Description	Exocet
 lee_cur_tools	Leeway and current tools	
 polar	Calculation of VMG and other boat speed performance data	
 wind_apparent	Back-calculation of the undamped apparent wind in the horizontal plane	
 wind_ground	Calculation of the ground wind, relative to the sea floor	
 wind_mastrot	Compensate the selected Mast Head Unit angle from the mast rotation and twist	
 wind_motion	Boat motion compensation on measured wind	
 wind_processor	Encode data to drive Processor HR with Pixel wind	
 wind_true	Calculate the original true wind in the horizontal plane	
 wind_upwash	Compensate the True Wind from various symmetrical errors	

## 11 Manta boxes detailed description

### 11.1 add\_delay



#### Description

Delays data streams.

This box accumulates a history of input streams and replays it with a delay. The history length is bounded.

#### Properties

- **Name** : Manta box name
- **Delay (s)** : Amount of time of the delay (s).
- **Outputs** : Number of data streams.
- **Dynamic delay** : Check this box to control the delay as an input.
- **WebApp** : Allow delay access from Exocet WebApp dashboards.
  - **Min / Max / Step** : Widget settings for window parameters.
- **Export parameters** : Outputs the current delay when it changes.

#### Inputs

- **Data *i*** : connect here the data streams containing variables to delay. Only one wire should be connected per Data input.
- **Delay** : connect here the delay parameter as input (s)
- **Reset** : reset dynamic and WebApp delay to initial value defined above

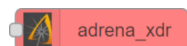
#### Outputs

- **Delayed data *i*** : Delayed data streams
- **Delay** : Delay parameter as output (s)

## WebApp controls

- **Delay (s)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change the *Delay (s)* value.

## 11.2 adrena\_xdr



### Description

Export data to Adrena navigation software. All received data are forwarded to Adrena via the XDR protocol.

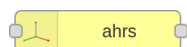
### Properties

- **Name** : Manta box name
- **IP** : Enter IP address of PC running Adrena
- **Port** : Enter the Adrena Xdr port

### Inputs

- Data to send to Adrena.

## 11.3 ahrs



### Description

This Manta box performs a calibration of the AHRS (Attitude and Heading Reference System) position and, if needed, apply magnetic corrections.

## Properties

- **Name** : Manta box name
- **Misroll** : Roll angle of sensor in boat frame
- **Mispitch** : Pitch angle of sensor in boat frame
- **Misyaw** : Yaw angle of sensor in boat frame
- **AHRS input mapping** : Fill Manta names corresponding to attitude, heading and magnetic deviation data
- **Magnetic variation input mapping** : Fill Manta name corresponding to magnetic variation

## Inputs

### AHRS raw data

Connect here one data stream containing raw sensor data (Euler angle of sensor in NED frame) \* *Heel (float, deg +/-180)* : Roll angle. Sign convention (-) port side down ; (+) port side up. \* *Trim (float, deg +/-180)* : Pitch angle. Sign convention : (-) bow down ; (+) bow up. \* *Heading (float, deg)* : Heading (Compass, Magnetic or True) \* *RollRate (float, deg/s)* \* *PitchRate (float, deg/s)* \* *YawRate (float, deg/s)* \* *Deviation (float, deg +/-180)* : Magnetic deviation from boat magnetic field.

### Magnetic variation

\* *Magnetic variation (float, deg +/-180)* : Magnetic variation from geographic position (also named magnetic declination).

## Outputs

**AHRS calibrated data** (Euler angle of boat in NED frame) : \* *Heel (float, deg +/-180)* : Roll angle. Sign convention (-) port side down ; (+) port side up. \* *Trim (float, deg +/-180)* : Pitch angle. Sign convention : (-) bow down ; (+) bow up. \* *Heading (float, deg T)* : True Heding. \* *RollRate (float, deg/s)* \* *PitchRate (float, deg/s)* \* *YawRate (float, deg/s)*

**Misalignments** (Euler angle of sensor in boat frame) : \* *Misroll (float, deg +/-180)* : Roll angle of sensor in boat frame \* *Mispitch (float, deg +/-180)* : Pitch angle of sensor in boat frame \* *Misyaw (float, deg +/-180)* : Yaw angle of sensor in boat frame

**Corrections** : corrections due to AHRS position calibration

## Notes

### Heading

Heading\_Magnetic = Heading\_Compass + Magnetic\_Deviation



Heading\_True = Heading\_Magnetic + Magnetic\_Variation

### Misalignments

Most applications will only have low angles on roll and pitch misalignment. If large angles on roll and pitch are expected ( $> 5^\circ$ ), user must consider the rotation composition order : roll, then pitch, then yaw.

### NED frame

The North-East-Down (NED) coordinate system with origin defined relative to the Earth's reference ellipsoid (World Geodetic System, 1984). This is the coordinate system we refer to in our everyday life. It is usually defined as the tangent plane on the surface of the Earth moving with the boat, but with axes pointing in different directions than the body-fixed axes of the boat. For this system the x axis points towards true North, the y axis points towards East while the z axis points downwards normal to the Earth's surface.

## 11.4 einstein



### Description

Decode Einstein altimeter UART & CAN protocol

### Properties

- **Name** : Manta box name
- **Protocol** : Select UART or CAN protocol
- **CAN ID** : Only for CAN protocol, enter CAN frame ID (hexadecimal)
- **Version ID** : Only for UART protocol, enter UART protocol version ID (byte 2 of data packet)

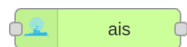
### Inputs

- **data\_in** : serial or can data stream

### Outputs

- **data** : altitude data
- **status** : altitude data status (SNR...)

## 11.5 ais



### Description

This box decodes and exports the AIS info from received NMEA0183 sentences. For now, only messages of type 1,2,3,5,18,19 and 24 are supported. The description of each message type is available [here](#).

### Properties

- **Name** : Manta box name

### Inputs

- **NMEA0183 VDO/VDM** : connect here the VDO/VDM output of the NMEA0183 decoding box.

### Outputs

- **Position report class A** : Return the full decoded class A position report (AIS message type 1,2,3)
- **Static and voyage related data** : Return the full decoded static and voyage data message (AIS message type 5)
- **Standard position report class B** : Return the full decoded standard position report class B (AIS message type 18)
- **Extended position report class B** : Return the full decoded extended position report class B (AIS message type 19)
- **Static data report** : Return the full decoded static data report (AIS message type 24)

## 11.6 alarm



### Description

This box exports two variables to inform if at least one data is on alarm state or on warning state. These variables can be used as entry value of *GPIO output box* to ring a buzzer when a data is on alarm.

Outputs Frequency : 5 Hz

## Properties

- **Name** : Manta box name
- **Group** : If a group is selected (from 1 to 12), only alarm/warning of the selected group are considered. If *Any* is selected, all alarm/warning are considered.

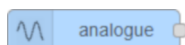
## Inputs

- **Acknowledge** (boolean) : acknowledge alarm/warning of the selected group on rising edge

## Outputs

- **Alarm** (boolean) : true if at least one data is on alarm with the selected group, else false.
- **Warning** (boolean) : true if at least one data is on warning with the selected group, else false.

## 11.7 analog



## Description

This Manta box handles analog to digital conversion. A 16-bit ADC (65536 discrete levels) converts voltage to digital data.

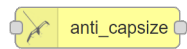
## Properties

- **Name** : Manta box name
- **Port** : Select the input source
- **Range** : Adjust the input voltage range to qualify the signals going into the ADC. Note that if the range is set to the minimum this will maximize the *signal to noise* ratio.
- **Sampling rate** : The number of input samples available per seconde. Warning : the output frequency could be affected by the number of *Analogue* boxes.
- **Add TS** : Add timestamp to output data flow

## Outputs

- Analog input value (*Float*, volt)

## 11.8 antiCapsize



### Description

This box is the angular piece of anti-capsize system.

It monitors a data and takes care of triggering actuators when this data exceeds one or more trigger thresholds.

Multiple TWA Ranges can be defined to have a different trigger threshold per TWA Range for the same actuator. For each TWA Range, an actuator trigger threshold list is defined. In this list, an actuator is associated with one trigger threshold per tack. Therefore the triggered actuator can be different on port and starboard tack for the same threshold.

The TWA range and tack selection can be either automatic by using TWA data or manual by using sliders on dashboard. To be in automatic selection, sliders have to be set on « auto » position. If the wind calculation is in default (TWA input error), it is possible to use the system in degraded mode by manually selecting the TWA range and tack. When the TWA range and tack are set in automatic selection and an error occurs on the TWA, the anti-capsize box continues to operate with the last automatically determined TWA range and tack.

This box requires an activation key to be activated.

### Properties

- **Name** : Manta box name.
- **Frequency** (*int, Hz*) : Execution Frequency

**Monitored data \* Comparison Sign** : Define if actuators are triggered when monitored data is « above or equal » thresholds or « below or equal » threshold. \* **Absolute Value** : If checked, the absolute value of the monitored data is compared to thresholds.

**TWA Ranges Definition \* TWA Range List** : Definition of the number of TWA ranges to consider and their names (max 4 TWA range). \* **TWA Limit N - Initial Value** (*int, deg*) : Initial value of the limit between 2 TWA range.

« TWA Limit N » = TWA limit between the « TWA Range N » and « TWA Range N+1 ».

**Actuator Trigger Thresholds \* Number of Actuators** (*int*) : Define the number of actuator to manage (max 30). \* **TWA Range Threshold lists** : Define the different trigger threshold to consider on each TWA rang (1 list by TWA range).

For each threshold the following parameters have to be set : \* **Threshold for** : Threshold name (use to defined WebApp widget name). \* **Initial Value** (*float*) : Initial threshold value. \* **Step Value** (*float*) : Step applied when incrementing or decrementing threshold value with WebApp widget. \* **Min Value** (*float*) : Minimum threshold value. \* **Max Value** (*float*) : Maximum threshold value. \* **Port Actuator** : Define the actuator to trigger on port tack. \* **Starbord Actuator** : Define the actuator to trigger on starboard tack.

**Input Variable Mapping** : Map the name of input variables.

## Inputs

**Monitored Data** (*float*) : Data whose value is compared to thresholds. Default timeout = 1s.

**TWA** (*float, deg +/-180*) : True Wind Angle (- : Port tack, + : Starboard tack). Default timeout = 1s.

**External Control** : Can be used to control WebApp Widgets from the manta graph. \* *Anti-Capsize On* (*bool*) : Control « AntiCapsizeOn » widget (True : ON, False : OFF) \* *TWA Range Selected* (*int*) : Control « TwaRangeSelected » widget. \* *Tack Selected* (*int*) : Control « TackSelected » widget. \* *TWA Limit N* (*int*) : Control « Twa Limit N » widget.

## Outputs

**Status** \* *Status* (*int*) : Anti-Capsize status. See note at the end for the meaning. \* *Status\_display* (*string*) : Anti-Capsize status color \* *ErrorStatus* (*int*) : Anti-Capsize error status (1 bit for each cause of error). See note at the end for the meaning. \* *TwaRangeName\_Status* (*int*) : TWA Range status (1 variable by TWA range). See note at the end for the meaning. \* *TwaRangeName\_Status\_display* (*string*) : TWA Range status color (1 variable by TWA range)

Output frequency : 1 Hz or data change

**Parameters** \* *TwaRangeSelected* (*int*) : TWA range selection slider position (0 : automatique selection, 1 : TWA Range 1, ...) \* *TackSelected* (*int*) : Tack selection slider position (0 : automatique selection, 1 : Port, 2 : Starboard) \* *Twa Limit N* (*int*) : TWA Limit N value (1 variable by TWA Limit) \* *TwaRangeName\_Threshold* (*float*) : Threshold value (1 variable by threshold)

Output frequency : 1 Hz or data change

**Actuators Control N** (1 port by actuator) \* *Trigger Actuator N* (*bool*) : True if at least one threshold associated to the actuator N is exceeded, False otherwise.

Output frequency : Execution Frequency defined in parameter.

**WebApp controls**

- **AntiCapsizeOn** (*bool*) : On/Off bistable button.
- **TwaRangeSelected** (*int*) : TWA range selection slider
- **TackSelected** (*int*) : Tack selection slider
- **Twa Limit N** (*int*) : TWA limit values (1 widget by Twa Limit).
- **TwaRangeName\_Threshold** (*float*) : Actuator threshold values (1 widget by threshold).

**Notes*****Anti-Capsize status***

Status	Status color	Meaning
0	« dimgray »	OFF
1	« lime »	ON with TWA Range & Tack automatic switch available
3	« orange »	ON without TWA Range or Tack automatic switch (TWA input error)
7	« red »	ERROR. See ErrorStatus to know the reason.

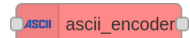
***Anti-Capsize ErrorStatus***

ErrorStatus Bit (little endian)	Meaning
Bit0 = 1	Monitored Data input error
Bit1 = 1	No TWA Range selected
Bit2 = 1	No Tack selected

***TWA Range status***

Status	Status color	Meaning
0	« dimgray »	OFF
1	« lime »	ON in standby
3	« orange »	ON and at least one actuator triggered

## 11.9 ascii\_out



### Description

Generic ASCII protocol encoder.

### Properties

- **Name** : Manta box name
- **Dec sep** : Select decimal separator for floating point values
- **List of fields** :
  - **Nature** : define if the field is a constant, filled from a manta variable or a checksum
  - **Value** : only for constants, value of the constant
  - **Hexa** : only for constants, check if value is in hexadecimal format (ex : 0x0D0A = CRLF)
  - **Variable** : only for variables, name of the manta variable
  - **Digits** : 0 to ignore, otherwise force digits number of the whole part
  - **Decimals** : only used for float, decimals number to use
  - **Type** : only for checksums, type of the checksum
  - **Begin** : only for checksums, position of first character to consider
  - **End** : only for checksums, position from the end of last character to consider

### Inputs

- **Variables** (*bool, scalar, string*). Only one wire should be connected.

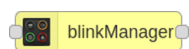
### Outputs

- **Sentence** : Should be connected to a serial or UDP output box.

### Notes

**NMEA0183** : To format an nmea0183 sentence, use constants to define header and comma separators between fields. The checksum must have the *LRC-8* type, and must begin at *1* and end at *-1*.

## 11.10 blinkManager



### Description

Ease of use Blink Powerkey Pro interface : \* Init Keypad \* Decode button state from N2000 \* Prepare led command to send through N2000

### Properties

- **Name** : Manta box name
- **Address** : Keypad CAN source address. (Several keypad can have the same address)
- **Model** : Select the number of buttons
- **Periodic** : If checked, active periodic transmission of key states at 1Hz
- **Mult. Led cmd** : If checked, allow multiple command at LED command input using *LedColor\_x* and *LedState\_x* to control Led of button X. Otherwise, support historical protocol where only one LED can be addressed using *KeyCode*, *LedColor* and *LedState*.
- **Day Led level (%)** : Enter LED brightness level in day mode
- **Day backlight (%)** : Enter backlight level in day mode
- **Night Led level (%)** : Enter LED brightness level in night mode
- **Night backlight (%)** : Enter backlight level in night mode
- **Mapping** : Enter your variables names of Led commands

### Inputs

- **N2k in** : Should be connected to port 61184 of a net2000 in box with type set as Powerkey
- **LED command** : Should be connected to your Led command management. Must contain :

Variable	Type	Description
<b>Key code</b>	Int	Key code ID (from 1 to 15)
<b>Led color</b>	Int	0 : nothing, 1 : red, 2 : green, 3 : blue, 4 : yellow, 5 : cyan, 6 : magenta, 7 : white, 8 : amber
<b>Led state</b>	Int	0 : Off, 1 : On, 2 : Blink
<b>Erase all</b>	Bool	Optional. If true, erase all before sending the command.

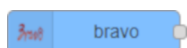


- **Night mode** (*bool*) : Connect a boolean to enable/disable night mode. Default is disabled.

## Outputs

- **N2k Led command** : Must be connected to port 61184 of a net2000 out box with type set as Powerkey
- **Button state** : Should be connected to your button state management.
- **N2k Led brightness** : Must be connected to port 61184 of a net2000 out box with type set as Powerkey
- **N2k backlight** : Must be connected to port 61184 of a net2000 out box with type set as Powerkey
- **N2k enable periodic** : Must be connected to port 61184 of a net2000 out box with type set as Powerkey
- **N2k period** : Must be connected to port 61184 of a net2000 out box with type set as Powerkey

## 11.11 bravo\_in



### Description

This Manta box imports data from Bravo 4 Sailing Instrument Processor.

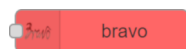
### Properties

- **Name** : Manta box name
- **Network** : Select network configuration. Press the *pencil* icon to create a new one. Enter the IP address and port number used to communicate with Bravo processor. Once created, it can be shared with a *Bravo out* box.
- **Bravo variables to import** : Press *New data* button. Fill the Bravo data name and choose the decimals to be used by Bravo to format the imported data.

## Outputs

- Specified Bravo variables to import

## 11.12 bravo\_out



### Description

This Manta box exports data to Bravo 4 Sailing Instrument Processor.

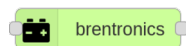
### Properties

- **Name** : Manta box name
- **Network** : Select network configuration. Press the *pencil* icon to create a new one. Enter the IP address and port number used to communicate with Bravo processor. Once created, it can be shared with a *Bravo in* box.
- **Bravo variables to export** : Press *New data* button. Fill the Exocet data name, the Bravo data name and choose the decimals to be used by Exocet to format the exported data. If decimal parameter is set to « Auto », then the number of decimals used to display the variable within Exocet Web App will be applied.

### Inputs

- Exocet variables to export to Bravo. (Multiple wires can be connected).

## 11.13 bretronics



### Description

Decode Bretronics battery status sent to CAN bus

### Properties

- **Name** : Manta box name
- **Reference** : Select the battery reference
- **Serial Number** : Enter the battery serial number as returned on CAN bus. In case of trouble, first check this parameter.

## Inputs

- **CAN\_input** : Must be connected to a canbus in box configured at 250 kbps

## Outputs

- **CAN\_output** : Must be connected to a canbus out box configured at 250 kbps
- **Battery\_Status** : Return the specified battery status :

Name	Units	Type
<b>Firmware_version</b>		float
<b>Firmware_patch</b>		integer
<b>Firmware_build</b>		integer
<b>AtRate</b>	Amperes	float
<b>AtRateTimeToFull</b>	Minutes	integer
<b>AtRateTimeToEmpty</b>	Minutes	integer
<b>AtRateOK</b>		bool
<b>Temperature</b>	Degrees Celcius	float
<b>Voltage</b>	Volts	float
<b>Current</b>	Amperes	float
<b>AvgCurrent</b>	Amperes	float
<b>MaxError</b>	Percent	integer
<b>RelStateOfCharge</b>	Percent	integer
<b>AbsStateOfCharge</b>	Percent	integer
<b>RemainingCapacity</b>	Amperes Hours	float
<b>FullChargeCapacity</b>	Amperes Hours	float
<b>RunTimeToEmpty</b>	Minutes	integer
<b>AvgTimeToEmpty</b>	Minutes	integer
<b>AvgTimeToFull</b>	Minutes	integer
<b>ChargingCurrent</b>	Amperes	float
<b>ChargingVoltage</b>	Volts	float

Name	Units	Type
<b>Flags</b>	<i>see below</i>	bit field
<b>CycleCount</b>		integer
<b>DesignCapacity</b>	Amperes Hours	float
<b>DesignVoltage</b>	Volts	float
<b>ManufactureDate</b>		integer
<b>SerialNumber</b>		integer

- **Flags** consolidates following Battery Status bit flags :
  - *Flags\_OverchargedAlarm*
  - *Flags\_TerminateChargeAlarm*
  - *Flags\_OverTemperatureAlarm*
  - *Flags\_TerminateDischargeAlarm*
  - *Flags\_RemainingCapacityAlarm*
  - *Flags\_RemainingTimeAlarm*
  - *Flags\_Initialization*
  - *Flags\_ChargeFETTest*
  - *Flags\_FullyCharged*
  - *Flags\_FullyDischarged*
  - *Flags\_ErrorCodes* : 0 ok, 1 busy, 2 reserved command, 3, unsupported command, 4 access denied, 5, overflow/underflow, 6 bad size, 7 unknown error

## Notes

Bren-Tronics battery should not be connected on a CAN bus where PGN 61184 is already using because conflicts can occurred.

## 11.14 button



## Description

User controlled box.

This Manta box is controlled from Exocet Web App dashboards via the « Button » widget. The boolean output of the box corresponds to the « Button » widget state.

### Properties

- **Name** : Manta box name
- **Save** : If selected, the button state is saved to be restored when Exocet is rebooted or Manta graph deployed.
- **Output Name** : Exported data name. Manta box name is used if not defined.
- **Period** : to send button state periodically. State is also sent at any user change. If **None** is selected, only value changes are sent.
- **Initial Value** : initial data value.
- **Ext control** : If selected, add inputs to externally control the button value.

### Inputs

- No Input connector, the input comes from Web App.
- **Bool** : If *Ext control* is selected, a boolean value can be received to set the data value.
- **Reset** : If *Ext control* is selected, a boolean rising edge can be received to return to the *Initial Value*.

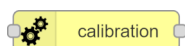
### Outputs

- **Output Name** : Boolean data value

### WebApp controls

- **Output Name (bool)** : On your dashboard, use a « Button » widget to switch the value.

## 11.15 calibration



### Description

The purposes of this Manta Box is :

- To convert measurement units (for example voltage, mV/V) to a physical unit (degrees C,...)
- To adjust the precision and accuracy of measurement equipments or to reduce bias in an instrument.
- To mux several data streams into one by selecting requested variables

Variables to be calibrated need to be added into the list.

### Properties

- **Name** : Manta box name
- **Pass through** : export all non calibrated variables.
- **Dynamic calibration** : activates external parameters inputs. Not available for linearization type.
- **WebApp** : allow parameters access from Exocet WebApp dashboards
- **Min / Max / Step** : (WebApp option only) scalar widget settings
- **Export parameters** : add an output connector to export current parameters (only for slope/offset)
- **Apply to all** : If checked, apply the same process to all variables of the input data stream.
- **Unit** : (Apply to all option only) Modify or remove the unit of calibrated variables.
- **Prefix** : (Apply to all option only) Use this field to prefix all output variables.
- **Suffix** : (Apply to all option only) Use this field to suffix all output variables.
- **Replace/by** : (Apply to all option only) Use these fields to search and replace a substring into all output variables.
- **List of variables to calibrate** (maximum 250) :
  - **Variable** : name of the variable to calibrate
  - **Type** : calibration type. See below for calibration types definition.
  - **Output Name** : Change the name of the calibrated variable.
  - **Unit** : Modify or remove the unit of the calibrated variable.
  - **Other properties depends on the calibration type.**

### Inputs

- **Data** : Connect here the data stream containing variables to calibrate.
- **slope/offset** : Connect scalar to configure calibration parameters.
- **table** : Connect table to configure calibration LUT.
- **Reset** : Connect here a boolean to reset dynamic and WebApp parameters to initial values defined above

## Outputs

- **Calibrated\_data** : A list of calibrated data, and non calibrated data if *Pass through* option is selected.
- **Parameters** : For Slope/Offset type, the list of current slope/offset

## WebApp controls

- **Output Name(slope)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *slope* value.
- **Output Name(offset)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *offset* value.
- **Output Name(X1)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *X1* value.
- **Output Name(Y1)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *Y1* value.
- **Output Name(X2)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *X2* value.
- **Output Name(Y2)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change *Y2* value.
- **Output Name(LUT)** (*floatTable*) : If *WebApp* is selected, on your dashboard, use a « Table input » widget to change *LUT* value.

## Notes

### Calibration types :

*Slope/Offset* : This type of calibration requires two parameters : the slope coefficient and the offset coefficient. Returns  $y=ax+b$ . Those coefficients are default values in case of dynamic calibration.

*Linearization* : This type of calibration requires 2 points to define a straight line. It calculates a slope and an offset, and then applies a standard slope/offset calibration.

*1D-LUT* : This type of calibration requires  $n$  points to define a transfert function (curve). These points are used as default values in case of dynamic calibration. Several outputs can be defined (see 1D-LUT format below). Interpolation could be linear, polynomial, trigonometric or spline. Be careful with some interpolation method, especially polynomial, which can create some unexpected behavior depending on defined points. Please, use the « Curves » widget in the WebApp to check the interpolation result.

**2D-LUT** : This type of calibration requires 2 input variables, n points for each, to define a transfert function (plane). These points are used as default values in case of dynamic calibration. Interpolation is linear.

#### 1D-LUT format :

The transfert function (curve) is defined by n pairs of x,y coordinates. The first column define x (must be in ascending order), the second define y. The column separator could be space or tab. Example :

```
0.0 1.0
10.0 1.5
20.0 1.5
```

If extrapolation is selected, first and last segment are used outside the LUT limits (to use very carefully if interpolation method is not linear). Otherwise, limits are defined as follows :

```
x < x[0] => y = y[0]
x > x[n] => y = y[n]
```

The 1D-LUT can define several curves for several outputs (number of y need to match with the *Nb output* field) :

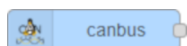
```
0.0 1.0 2.0
10.0 1.5 2.0
20.0 1.5 3.0
```

#### 2D-LUT format :

The transfert function (plane) is defined by 2 input variables in a x,y,z coordinates. The first column define x (*Variable*, must be in ascending order), the first line define y (*Variable 2*, must be in ascending order), the intersections define results z. The column separator could be space or tab. The first line first column will be ignored but must be a valid number. Example :

```
0 45.0 90.0
0.0 1.0 1.0
10.0 1.0 1.5
20.0 1.5 2.0
```

### 11.16 canbus\_in





## Description

Get data from the specified CAN Bus.

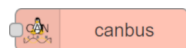
## Properties

- **Name** : Manta box name
- **CAN port** : CAN port configuration box. Press the pencil icon to create a new one. Fill the CAN port parameters, then press the *Add* button. Once created, it can be shared with other CAN in our out boxes.
- **Add TS** : Add timestamp to output data flow
- **Filter** : If selected, export only can frames with *CAN ID* header.
- **CAN IDs to filter out** : Only if *Filter* is selected, list of CAN frame IDs to export (hexadecimal)

## Outputs

- **can** (*CanFrame*) : received CAN data

## 11.17 canbus\_out



## Description

Send data to the specified CAN Bus.

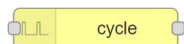
## Properties

- **Name** : Manta box name
- **CAN port** : CAN port configuration box. Press the pencil icon to create a new one. Fill the CAN port parameters, then press the *Add* button. Once created, it can be shared with other CAN in our out boxes.

## Inputs

- **can** (*CanFrame*) : data to send

## 11.18 cycle



### Description

Execute a periodic digital signal when its input goes to True state.

### Properties

- **Name** : Manta box name
- **Output Name** : Exported data name. Manta box name is used if not defined.
- **Period** : Signal periodicity in seconds (from 1s to 15s)
- **Duty cycle** : Signal duty cycle (from 0% to 100%). Default step is 10%, use *high resolution* option to decrease step to 1%.
- **High resolution** : Permit to increase duty cycle resolution from 10% to 1%.
- **Max repetition** : Number of cycles to execute when input is triggered (from 0 to 10, 0 = infinite)
- **Complete** : If true, execute max repetition cycles even if input return to False state. If false, stop to execute the digital signal when input return to False state.
- **Ext control** : If selected, add inputs to externally control the signal parameters.

### Inputs

- **Trigger** : A boolean to activate the periodic digital signal
- **Ext control** : If *Ext control* is selected, scalar can be received to set the signal period, duty cycle and max repetition

Name	Description
<b>Period</b>	Signal periodicity in seconds (from 1 to 15, step 1)
<b>DutyCycle</b>	Signal duty cycle (from 0 to 100, step 10 or 1)
<b>MaxRepetition</b>	Number of cycles to execute when input is triggered (from 0 to 10, 0 = infinite)

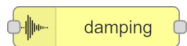
- **Reset** : If *Ext control* is selected, a boolean rising edge can be received to return to the initial signal parameters

**Outputs :**

- **Signal** : The periodic digital signal when input is triggered. Data frequency is 0.1Hz in idle state, 10Hz or 100Hz when input is triggered depending on *high resolution* option.
- **Parameters** : Export the current signal parameters

**WebApp controls**

- **Period** (*int*) : On your dashboard, use a « Set number » or « Slider » widget to change the *Period* value.
- **DutyCycle** (*int*) : On your dashboard, use a « Set number » or « Slider » widget to change the *Duty cycle* value.
- **MaxRepetition** (*int*) : On your dashboard, use a « Set number » or « Slider » widget to change the *Max repetition* value.

**11.19 damping****Description**

The purposes of this Manta Box is to apply a digital infinite-impulse-response (IIR) filter on all or selected variables.

**Properties**

- **Name** : Manta box name
- **Dynamic** : If selected, add inputs to control cutoff from the Manta graph
- **Export params** : add an output connector to export current parameters
- **WebApp** : Allow cutoff parameters access from Exocet WebApp dashboards
- **Min / Max / Step** : (*WebApp* option only) widget settings for cutoff parameters
- **Design** : Design of damping to apply (see *Notes* below)
- **Attenuation (dB)** : (*Chebyshev* option only) Minimum stopband attenuation in dB
- **Freq div** : Divisor factor to reduce input sampling frequency to output sampling frequency (see *Notes* below)
- **Apply to all** : If checked, apply the same process to all variables of the input data stream
- **Pass through** : (Unchecked *Apply to all* option only) Export all non damped variables

- **Auto suffix** : (*Apply to all* option only) Check to add an automatic suffix to all variables, else use the input field to define a custom suffix
- **Type** : Type of damping to apply (see *Notes* below)
- **Setting type** : Type of the damping setting : *Frequency (Hz)* or *Time Constant (s)*. Both are linked with  $\text{Time Constant} = 1 / (2 * \text{PI} * \text{Cutoff Frequency})$ . The cutoff frequency corresponds to a 3dB attenuation.
- **Setting value** : Value of the damping setting relative to the selected *Setting type* (see *Notes* below)
- **List of variables to damp** (maximum 250) :
  - **Var type** : Type of variable : scalar or complex (see *Notes* below)
  - **Var\_in** : Name of the scalar variable to damp
  - **Var\_out** : Name of the damped scalar variable
  - **Mod\_in** : Name of the modulus part of complex variable to damp
  - **Mod\_out** : Name of the modulus part of damped complex variable
  - **Arg\_in** : Name of the argument part of complex variable to damp (rad or °, default is °)
  - **Arg\_out** : Name of the argument part of damped complex variable

## Inputs

- **Data** : Connect here the data stream containing variables to damp. Only one wire must be connected on Data input.
- **cutoff** : Connect here one (or two) scalar to configure the « Setting Value(s) » dynamically. Unit depends on « Setting Type » selection.
- **Reset** : Connect here a boolean to reset dynamic and WebApp parameters to initial values defined above.

## Outputs

- **Damped\_data** : A list of damped data, and non damped data if *Pass through* option is selected.
- **Parameters** : The list of current « Setting Values »

## WebApp controls

- **Output Name(Hz or s)** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change the *Settings Value*.

## Notes

### Designs

Name	Delay for lowpass filter	Max cut-off frequency	Roll-off
<b>Butterworth order 1</b>	Equal to <i>Time Constant</i>	No limit	20 dB / decade
<b>Butterworth order 2</b>	Equal to 1.4 <i>Time Constant</i>	50 Hz	40 dB / decade
<b>Chebyshev type II order 2</b>	Less than 1.4 <i>Time Constant</i>	50 Hz	Slightly more than 40 dB / decade until <i>Attenuation</i> is reached

### Types

- **Low Pass** : passes signals with frequency lower than selected cutoff frequency, and attenuates signals with higher frequencies.
- **High Pass** : passes signals with frequency higher than selected cutoff frequency, and attenuates signals with lower frequencies.
- **Band Pass** : passes signals with frequency within range, and attenuates signals with outside frequencies.
- **Band stop** : attenuates signals with frequency within range, and passes signals with outside frequencies.

### Frequency Divider

Used to re-sample output signal relative to input signal sampling frequency. If high frequencies have been attenuated, there is no reason to keep the input sampling frequency. Minimum output sampling frequency should be twice the max cutoff frequency (i.e. 2Hz in case of cutoff at 1Hz).

### Setting value

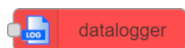
For 2nd order filters, an internal sampling is done at 100Hz. The filter cutoff frequency must be lower than the internal sampling frequency divided by two. Hence the cutoff frequency cannot be beyond 50Hz. For the Butterworth order 1 filter, there is no such limitation.

### Complex variable type

The box converts a vector defined by *Modulus* and *Argument* from polar coordinate to a complex number in a form  $z=a+ib$ . Damping is applied to the complex form and then convert back to a vector

in polar coordinate vector.

## 11.20 datalogger



### Description

The purposes of this Manta Box is to store received data in permanent memory. Received data are stored within Exocet SD card using PSM file format. PSM is a compressed file format that is proprietary. Stored data can be exported to a CSV file via the PSM decoder windows application or from the Exocet Web App.

The datalogger :

- can handle data type *Boolean*, *Integer*, *Float*, *String* and *CanFrame*.
- cannot handle *Opaque* data (from Serial, UDP, ...) nor *FloatArray* or *FloatTable*.

Multiple datalogger boxes can be placed in the Manta graph, but they all will store data within the same PSM file.

The datalogger can also send live data thanks to the Cloud application.

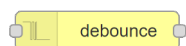
### Properties

- **Name** : Manta box name
- **Full log** : If checked, every data will be logged. If unchecked, data is only logged on change.  
**IMPORTANT NOTE** : Using **Full log** can significantly increase the size of PSM files and cloud transmissions if data to log have a high frequency. It is recommended to use this option only during setup/debug phase and not in production.
- **Decimals** : Number of decimals used to determine if a float data has changed, and to print this data into CSV file when *Exocet* float format is selected in PSM Decoder.
- **Specific decimals** : List of variables that need a specific *Decimals* parameter
- **Freq limit** : Limit logged data frequency
- **Cloud** : Enable Cloud transmission
- **Cloud conf** : Cloud configuration box. Press the pencil icon to create. Only one configuration is allowed, it is shared with other datalogger boxes.
- **Variables** : Select variables to send to the Cloud application. Note that the Manta graph must have been deployed a first time after adding the *Datalogger* box and connecting it, to populate the variables list.

## Inputs

- **DATA** : Connect here data to log. Multiple wires can be connected.
- **Enable** (*bool*) : Connect a boolean to enable/disable the datalog. Default is enable.

### 11.21 debounce



## Description

Execute a debouncing filter on specified variables. Rising and Falling edge could be separately configured.

## Properties

- **Name** : Manta box name
- **List of variables to process** :
  - **Variable** : name of the input variable to process
  - **Output Name** : Exported data name. Input variable name is used if not defined
  - **Rising** : Delay in milliseconds to filter out any glitches on input rising edge (step is 50ms)
  - **Falling** : Delay in milliseconds to filter out any glitches on input falling edge (step is 50ms)
  - **Invert** : Invert the output digital signal

Input variable type can be boolean, a scalar or a string. Scalar (Integer or Float) with value equal or greather than 1 is considered as true, else it is considered as false. String containing either « Yes » or « On » or « Enabled » or « 1 » is considered as true, else it is considered as false.

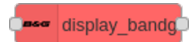
## Inputs

- **Data** : connect here the data stream containing variables to process

## Outputs :

- **Output** : processed data

## 11.22 display\_bandg



### Description

This box is used to send custom variables to B&G displays.

### Properties

- **Name** : Manta box name.
- **Emission Period** : Data emission period (in seconds).
- **List of variables to display** :
  - **Variable** : Name of the input variable to display
  - **Channel** : Select channel to use
  - **Long caption** : Data label used in display menu (max 16 characters). If empty, the name of the input variable truncated at 16 characters is used.
  - **Short caption** : Data label used on display screen (max 8 characters). If empty, the name of the input variable truncated at 8 characters is used.
  - **Decimals** : Select decimal number to display
- **Known User Vars on CAN bus** :
  - **CAN Port** : Select CAN bus port of net2000 out box
  - **User Vars** : Channel - Long caption - Short caption - Decimals

### Inputs

Variables to display (*Integer or Decimal*). Multiple wires can be connected.

### Outputs

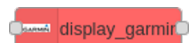
- **PGN 130833** : Must be connected to port 130833 of a net2000 out box with type set as User var
- **PGN 130824\_x** : Must be connected to port 130824 of a net2000 out box with type set as User var



## Notes

Use input timeout to display dashes.

### 11.23 display\_garmin



## Description

This box is used to send custom variables to Garmin displays.

## Properties

- **Name** : Manta box name.
- **Emission Period** : Data emission period (in seconds).
- **List of variables to display** :
  - **Variable** : Name of the input variable to display
  - **Channel number** : Select channel to use
  - **Format** : Select desired format
  - **Display name** : Data label used on display screen (max 10 characters). If empty, the name of the input variable truncated at 10 characters is used.
  - **Timeout** : Enter desired timeout in second. Default is 0 (no timeout).

## Inputs

Variables to display (*Boolean, Scalar or Time*). Multiple wires can be connected.

## Outputs

- **126720\_X** : Must be connected to port 126720 of a net2000 out box with type set as User var

## Notes

Use input timeout to send dashes.

***Formats :***

Format	Description
Invalid	Display dashes [---]
Unsigned	Display scalar as unsigned integer [0 :4294967295]
Signed	Display scalar as signed integer [- 2147483648 :2147483647]
Float	Display scalar as float [- XXXX.XXXX :XXXX.XXXX]
Boolean	Display boolean with translated string True/False or 1/0
Unsigned angle	Display scalar as unsigned angle, with degree symbol [0° :360°]
Signed angle	Display scalar as signed angle, with degree symbol [-180° :+180°]

Format	Description
3 digits angle	Display scalar as unsigned angle padded with leading zeros to 3 digits, with degree symbol [000°,001°:360°]
P/S angle	Display scalar as signed angle, with degree symbol and P/S indicator, P sent as negative [180°P :+180°S]
Temperature	Display scalar as float (convert from Celcius), with degree symbol [-12°,35°]
Length	Display scalar as float (convert from meters)
Depth	Display scalar as float (convert from meters)
Speed	Display scalar as float (convert from nautical knots)

Format	Description
Pressure	Display scalar as float (convert from millibar)
Volume	Display scalar as float (convert from meters cubed)
Flow	Display scalar as float (convert from meters cubed per hour)
Voltage	Display scalar as float (convert from Volts)
Percent	Display scalar as unsigned integer, with percent symbol
Port / Starboard	Display boolean with translated string Port/Starboard (Port is True)
On / Off	Display boolean with translated string On/Off (On is True)

Format	Description
Yes / No	Display boolean with translated string Yes/No (Yes is True)
Time	Display scalar as time HH :MM :SS (convert from Seconds) [00 :00 :00 to 99 :59 :59]

## 11.24 display\_nke



### Description

This box is used to send custom variables to NKE displays.

### Properties

- **Name** : Manta box name.
- **Emission Period** : Data emission period (in seconds).
- **List of variables to display** :
  - **Variable** : Name of the input variable to display.
  - **Channel number** : 1 to 8 or 1 to 30 depending on NKE display.
  - **Format** : Display format. See below for the format definition.
  - **Display name** : Data label of the display (max 10 characters). If empty, the name of the input variable truncated at 10 characters is used.
  - **Overwrite unit** : Checkbox to overwrite the unit label of the display.
    - \* Unchecked : The unit name of the input variable truncated at 7 characters is used.

\* Checked : The *display unit* field is used.

- **Display unit** : Unit label of the display (max 7 characters).

## Inputs

Variable to display (*Integer or Decimal*). Multiple wires can be connected.

## Outputs

- **data** (*Opaque*) : NKE proprietary NMEA0183 sentence. The output connector must be connected to a serial or udp\_out box.

## Notes

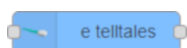
**Output Frequency** : It is defined by the emission period.

### Formats :

Format label	Data value
XX_XX	00.00 to 99.99
XXXX	0000 to 9999
XX	00 to 99
XXX_X	000.0 to 999.9
XDIR	-179° to 180°
XANG	000° to 359°
X_XXX	0.000 to 9.999
XCHRONO	00 :00 to 99 :59
XX_X	00.0 to 99.9
SXXX	-999 to 999
SXX_X	-99.9 to 99.9
SX_XX	-9.99 to 9.99
XXPC	00% to 99%

Format label	Data value
XREND	0000 to 9999
XLONG	0L to 999L

## 11.25 e\_telldatales



### Description

This box exports the angles for a set of e-telldatales.

### Properties

- **Name** : Manta box name
- **List of E-telldatales to export** : define a list of e-telldatales to export. For each, define :
  - **ID** : identifier. *nmea0183\_decode* Manta box with « *PENON* » prefix can be used to list all the e-telldatales identifier available.
  - **Ax** : Constant to calibrate angle data. Update this value on e-telldatales manufacturer recommendation only, else keep default value.
  - **B** : Constant to calibrate angle data. Update this value on e-telldatales manufacturer recommendation only, else keep default value.

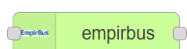
### Inputs

- **nmea0183 data** : nmea0183 stream from serial or UDP.

### Outputs

- One output per listed e-telldatale. Each output exports the e-telldatale angle.

## 11.26 empirbus





## Description

Decode EmpirBus user data sent over CAN bus.

## Properties

- **Name** : Manta box name
- **Instance** : unique instance to distinguish / route the data
- **Data Model** : Data model of Application Specific Data
- **List of data to import** :
  - **Field** : EmpirBus user data field
  - **Output name** : name of the imported variable
  - **Unit** : unit of the imported variable

## Inputs

- **PGN 65280** : Must be connected to port 65280 of a net2000\_in box with type set as Energy

## Outputs

- A list of decoded data according to the list defined in properties

## Note

**Application example** : Below is an example of EmpirBus output configuration. It uses Data Model 2 to export a Dimmer value on User Data Word 1, and an AC fan status on User Data Bit 5 :8.

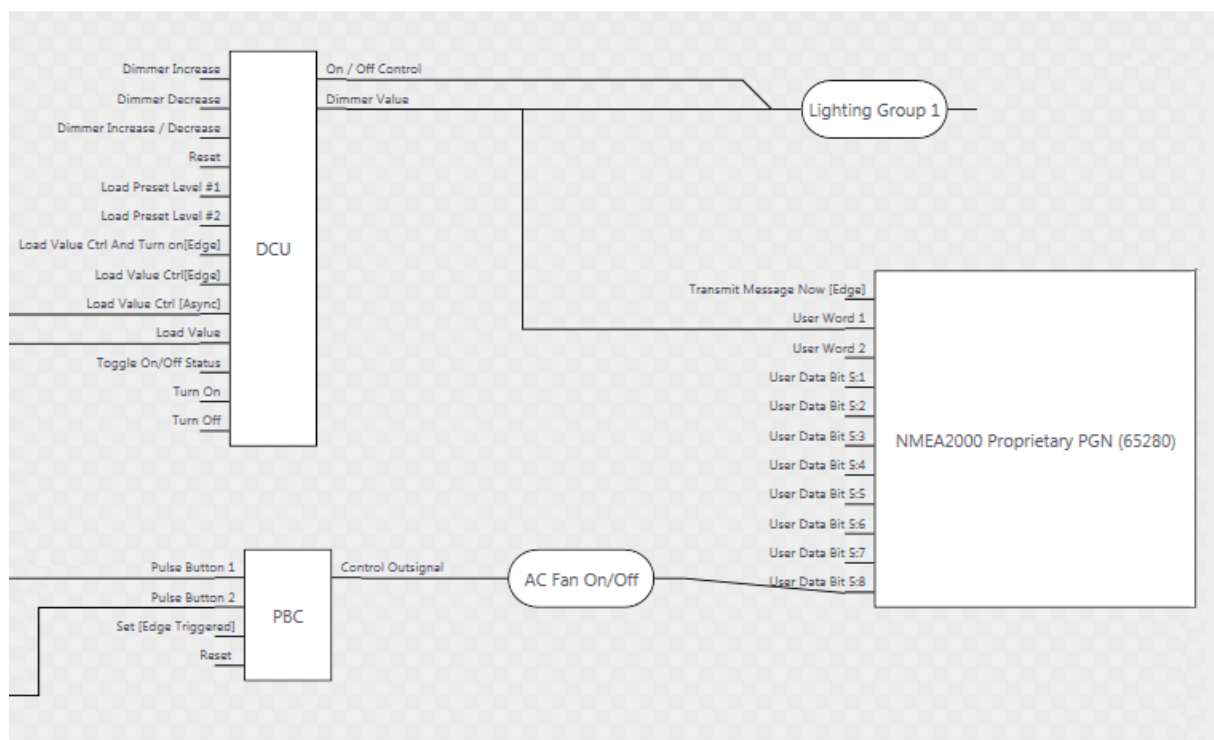
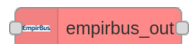


FIG. 79

## 11.27 empirbus\_out



### Description

Encode EmpirBus user data to send over CAN bus.

### Properties

- **Name** : Manta box name
- **Instance** : unique instance to distinguish / route the data
- **Data Model** : Data model of Application Specific Data
- **List of data to export** :
  - **Variable** : name of the variable to export
  - **Field** : EmpirBus user data field

## Inputs

- Variables to export (*bool, int or float*). Multiple wires can be connected.

## Outputs

- PGN 65280** : Must be connected to port 65280 of a net2000\_out box with type set as Energy

## Note

**Application example** : Below is an example of EmpirBus input configuration. It uses Data Model 2 to import a Load value on User Data Word 1, and a boolean command on User Data Bit 5 :8.

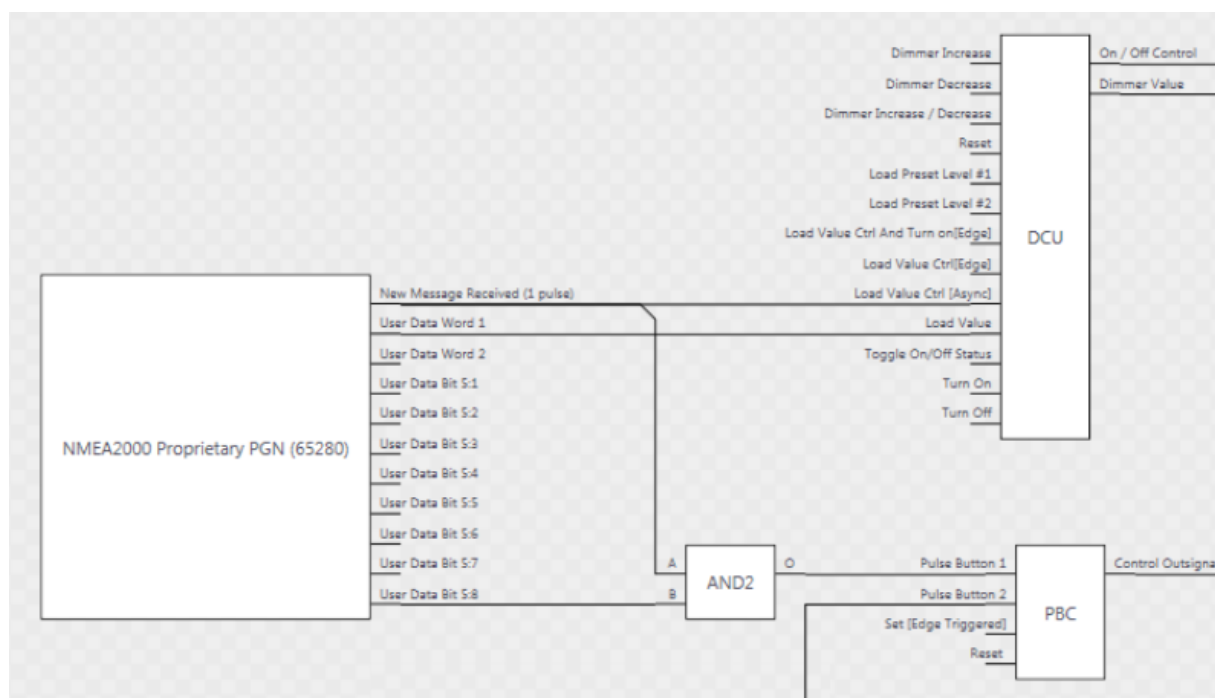
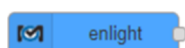


FIG. 80

## 11.28 enlight



## Description

Export variables such as microstrains or temperatures from Enlight, the Micron Optics optical interrogator PC software. It is compatible with all the Optics interrogators.

Notes :

- The grouping notion of Enlighth is not taken into account for now
- The variables created from the *Sensor* panel are exported without unit
- The *remote command interface* parameter must be checked within *settings* panel

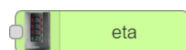
## Properties

- **Name** : Manta box name
- **IP** : IP address of PC running **Enlight**

## Outputs

- A set of variables such as microstrains or temperatures according to the configuration file

### 11.29 eta



## Description

Decode ETA Power distribution Systems sent over CAN bus.

Notes :

- ETA system must first be configured using PowerPlex software to periodically transmit load outputs current and analog inputs voltage over CAN bus.
- Current value feature is not available for the 1 Ampere load outputs. Only the 8A and 25A load outputs are able to transmit it. To worlaround this, a constant current value can be associated with a memory flag ID. If the 1A current output is linked with a this memory flag, then the constant value will be exported when 1A output is switched on.

## Properties

- **Name** : Manta box name
- **ID** : ETA system ID.
- **List of ETA data to export** : define a list of data to export. For each data, select :
  - **Type** :
    - **Current Out 1A** : to send a constant value when output is switched on.
    - **Current Out 8A** : to get current transmitted by a 8 Amperes load output.
    - **Current Out 25A** : to get current transmitted by a 25 Amperes load output.
    - **Analog In 0-10V** : to get analogic input voltage.
  - **ID** : load output ID or analogic input ID or memory flag ID
  - **out name** : can be used to define a name for the output. If not set, a default name will be defined.
  - **Value** : constant value to export when 1 A output is switched on. Else 0 ampere value is exported.

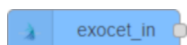
## Inputs

- **canin** : can bus input to receive data from ETA system.

## Outputs

- A list of *analogID* or *currentID*, respectively in Volts or Amperes according to ETA box configuration.

### 11.30 exocet\_in



## Description

This Manta box imports data from another Exocet Blue, Silver or Gold.

## Properties

- **Name** : Manta box name
- **Local port** : Enter here the local input port to receive data. You need to customize the default value if you use several Exocet input boxes.

- **Remote IP** : Enter here the emitter IP address.
- **Remote port** : Enter here the emitter output port. You need to customize the default value if there is several Exocet output boxes.
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards

## Outputs

- Imported Exocet variables

## Notes

Maximum number of variables is limited to 1024.

### 11.31 exocet\_out



## Description

This Manta box exports data to another Exocet Blue, Silver or Gold.

## Properties

- **Name** : Manta box name
- **Local port** : Enter here the local output port to send data. You need to customize the default value if you use several Exocet output boxes.
- **Remote IP** : Enter here the receiver IP address.
- **Remote port** : Enter here the receiver input port. You need to customize the default value if there is several Exocet input boxes.
- **Frequency** : Select the data emission frequency. It should be enough to ensure a good performance, but slow enough to not overload the receiver.
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards

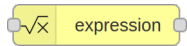
## Inputs

- Exocet variables to export. (Multiple wires can be connected).

## Notes

Maximum number of variables is limited to 1024.

### 11.32 expression



## Description

This Manta Box uses the C++ Mathematical Expression Toolkit (ExprTk) to support fundamental arithmetic operations, functions and processes. Each expression must finish by a semicolon « ; » (see *Notes* below for examples and more details), including loops that finish by a closing bracket « } ».

## Properties

- **Name** : Manta box name
- **Expression** : write here the expression to process. See *Notes* below for help.
- **List of symbols (maximum 100)** :
  - **Symbol** : symbol used in the expression
  - **Nature** : define if the symbol is a constant, an input/output variable, or an internal variable
  - **Value** : only for constants, value of the constant (scalar only)
  - **VarName** : name of the variable used in the Manta graph
  - **Direction** : input for variables to look for on input data stream, output for variables to export
  - **VarType** : type of the variable used in the Manta graph (note that inside this box Integer and Boolean are managed as floating point number, and CanFrame are managed as string)
  - **Init** : Check box to initialize input variable, and fill field with initial value to use. Otherwise, initialization of variable is checked before expression execution.
  - **Unit** : only for output direction, unit applied to the exported variable
  - **Type** : only for internal variables, select if the variable is a Float, a String or a Vector (of Float)
  - **Init** : only for internal variables, fill field with initial value to use. For a Vector, all elements are set to this value.
  - **Size** : only for internal variables of type Vector, size of the vector.

**Inputs**

- **Data** : connect here the data stream containing variables required to process the expression.

**Outputs**

- **Result** : export output variables defined in the list of symbols, depending on assignments of defined expression.

**Notes****Example**

```
var bsp := AWS * sin(AWA);
if (bsp > 35.0, debug := 'Too fast',
if (bsp < 1.0, debug := 'Too slow',
debug := 'cool'));
```

With : \* AWS, AWA : symbols of input variables of type double \* bsp : symbol of a local variable \* debug : symbol of an output variable of type string

**Reserved symbols** Due to the internal processing, the following words are reserved and so cannot be used as symbol : *abs, acos, acosh, and, asin, asinh, atan, atanh, atan2, avg, break, case, ceil, clamp, continue, cos, cosh, cot, csc, default, deg2grad, deg2rad, equal, erf, erfc, exp, expm1, false, floor, for, frac, grad2deg, hypot, iclamp, if, else, ilike, in, inrange, like, log, log10, log2, logn, log1p, mand, max, min, mod, mor, mul, ncdf, nand, nor, not, not\_equal, null, or, pow, rad2deg, rep, repeat, return, root, round, roundn, sec, sgn, shl, shr, sin, sinc, sinh, sqrt, sum, swap, switch, tan, tanh, true, trunc, until, var, while, xnor, xor, &, |*

**Disable output** A special internal variable named « *rep* » can be used to disable data export on output. If *rep* := *true*, data is exported (default), if *rep* := *false* no data is exported.

```
if (bsp < 1.0, rep := false, rep := true);
```

**Arithmetic & Assignment Operators**



OPERATOR	DEFINITION
+	Addition between x and y. (eg : $x + y$ )
-	Subtraction between x and y. (eg : $x - y$ )
*	Multiplication between x and y. (eg : $x * y$ )
/	Division between x and y. (eg : $x / y$ )
%	Modulus of x with respect to y. (eg : $x \% y$ )
^	x to the power of y. (eg : $x ^ y$ )
:=	Assign the value of x to y. (eg : $y := x$ )
+=	Increment x by the value of the expression on the right hand side. (eg : $x += y$ )
-=	Decrement x by the value of the expression on the right hand side. (eg : $x -= y$ )
*=	Assign the multiplication of x by the value of the expression on the righthand side to x. (eg : $x *= y$ )    /=   Assign the division of x by the value of the expression on the right-hand side to x. (eg : $x /= y$ )    %=   Assign x modulo the value of the expression on the right hand side to x. (eg : $x \% = y$ )

### Equalities & Inequalities

OPERATOR	DEFINITION
== or =	True only if x is strictly equal to y. (eg : $x == y$ )
<> or !=	True only if x does not equal y. (eg : $x <> y$ or $x != y$ )
<	True only if x is less than y. (eg : $x < y$ )
<=	True only if x is less than or equal to y. (eg : $x <= y$ )
>	True only if x is greater than y. (eg : $x > y$ )
>=	True only if x greater than or equal to y. (eg : $x >= y$ )

### Boolean Operations

OPERATOR	DEFINITION
true	True state or any value other than zero (typically 1).
false	False state, value of exactly zero.
and	Logical AND, True only if x and y are both true. (eg : x and y)
mand	Multi-input logical AND, True only if all inputs are true. Left to right short-circuiting of expressions. (eg : mand(x > y, z < w, u or v, w and x))
mor	Multi-input logical OR, True if at least one of the inputs are true. Left to right short-circuiting of expressions. (eg : mor(x > y, z < w, u or v, w and x))
nand	Logical NAND, True only if either x or y is false. (eg : x nand y)
nor	Logical NOR, True only if the result of x or y is false (eg : x nor y)
not	Logical NOT, Negate the logical sense of the input. (eg : not(x and y) == x nand y)
or	Logical OR, True if either x or y is true. (eg : x or y)
xor	Logical XOR, True only if the logical states of x and y differ. (eg : x xor y)
xnor	Logical XNOR, True iff the biconditional of x and y is satisfied. (eg : x xnor y)

### General Purpose Functions

FUNCTION	DEFINITION
abs	Absolute value of x. (eg : abs(x))
avg	Average of all the inputs. (eg : avg(x,y,z) == (x + y + z) / 3)
ceil	Smallest integer that is greater than or equal to x. (eg : ceil(x))
clamp	Clamp x in range between r0 and r1, where r0 < r1. (eg : clamp(r0,x,r1))
equal	Equality test between x and y using normalised epsilon
erf	Error function of x. (eg : erf(x))
erfc	Complimentary error function of x. (eg : erfc(x))
exp	e to the power of x. (eg : exp(x))
expm1	e to the power of x minus 1, where x is very small. (eg : expm1(x))
floor	Largest integer that is less than or equal to x. (eg : floor(x))

FUNCTION	DEFINITION
frac	Fractional portion of x. (eg : frac(x))
hypot	Hypotenuse of x and y (eg : hypot(x,y) = sqrt(x*x + y*y))
iclamp	Inverse-clamp x outside of the range r0 and r1. Where $r0 < r1$ . If x is within the range it will snap to the closest bound. (eg : iclamp(r0,x,r1))
inrange	In-range returns « true » when x is within the range r0 and r1. Where $r0 < r1$ . (eg : inrange(r0,x,r1))
log	Natural logarithm of x. (eg : log(x))
log10	Base 10 logarithm of x. (eg : log10(x))
log1p	Natural logarithm of 1 + x, where x is very small. (eg : log1p(x))
log2	Base 2 logarithm of x. (eg : log2(x))
logn	Base N logarithm of x. where n is a positive integer. (eg : logn(x,8))
max	Largest value of all the inputs. (eg : max(x,y,z))
min	Smallest value of all the inputs. (eg : min(x,y,z))
mul	Product of all the inputs (eg : mul(x,y,z) == (x * y * z))
ncdf	Normal cumulative distribution function. (eg : ncdf(x))
nequal	Not-equal test between x and y using normalised epsilon
pow	x to the power of y. (eg : pow(x,y) == x ^ y)
root	Nth-Root of x. where n is a positive integer. (eg : root(x,3) == x^(1/3))
round	Round x to the nearest integer. (eg : round(x))
roundn	Round x to n decimal places (eg : roundn(x,3)) where n > 0 and is an integer. (eg : roundn(1.23456,2) == 1.23)
sgn	Sign of x, -1 where $x < 0$ , +1 where $x > 0$ , else zero. (eg : sgn(x))
sqrt	Square root of x, where $x \geq 0$ . (eg : sqrt(x))
sum	Sum of all the inputs. (eg : sum(x,y,z) == (x + y + z))
swap, <=>	Swap the values of the variables x and y and return the current value of y. (eg : swap(x,y) or x <=> y)
trunc	Integer portion of x. (eg : trunc(x))

**Trigonometry Functions**

FUNCTION	DEFINITION
acos	Arc cosine of x expressed in radians. Interval [-1,+1] (eg : acos(x))
acosh	Inverse hyperbolic cosine of x expressed in radians. (eg : acosh(x))
asin	Arc sine of x expressed in radians. Interval [-1,+1] (eg : asin(x))
asinh	Inverse hyperbolic sine of x expressed in radians. (eg : asinh(x))
atan	Arc tangent of x expressed in radians. Interval -1,+1
atan2	Arc tangent of (x / y) expressed in radians. [-pi,+pi] (eg : atan2(x,y))
atanh	Inverse hyperbolic tangent of x expressed in radians. (eg : atanh(x))
cos	Cosine of x. (eg : cos(x))
cosh	Hyperbolic cosine of x. (eg : cosh(x))
cot	Cotangent of x. (eg : cot(x))
csc	Cosecant of x. (eg : csc(x))
sec	Secant of x. (eg : sec(x))
sin	Sine of x. (eg : sin(x))
sinc	Sine cardinal of x. (eg : sinc(x))
sinh	Hyperbolic sine of x. (eg : sinh(x))
tan	Tangent of x. (eg : tan(x))
tanh	Hyperbolic tangent of x. (eg : tanh(x))
deg2rad	Convert x from degrees to radians. (eg : deg2rad(x))
deg2grad	Convert x from degrees to gradians. (eg : deg2grad(x))
rad2deg	Convert x from radians to degrees. (eg : rad2deg(x))
grad2deg	Convert x from gradians to degrees. (eg : grad2deg(x))

**String Processing**

FUNCTION	DEFINITION
=, ==, !=, <>, <=, >=, <, >	All common equality/inequality operators are applicable to strings and are applied in a case sensitive manner. In the following example x, y and z are of type string. (eg : (x <= « AbC ») and (y != « aaa ») and (z == x))
in	True only if x is a substring of y. (eg : « abc » in « abcdef »)
like	True only if the string x matches the pattern y in a case sensitive manner. Available wildcard characters are « * » and « ? » denoting zero or more and zero or one matches respectively. (eg : « abcdef » like « a?cf' »)    <i>ilike</i>   <i>True only if the string x matches the pattern y in a case insensitive manner. Available wildcard characters are '*' and '?' denoting zero or more and zero or one matches respectively. (eg : 'ABCDEF' ilike 'a?cf' »)</i>
[r0 :r1]	The closed interval [r0,r1] of the specified string, where r0 < r1. (eg : with x := « abcdef », x[1 :3] == « bcd », x[:2] == « abc », x[3 :] == « def », x[: ] == « abcdef ») Note : In case of fractional components, truncation will be performed. (eg : 1.67 -> 1)
:=	Assign the value of x to y. (eg : y := x, y := « abc »)
+	Concatenation of x and y. (eg : x + y, x + « abc »)
+=	Add to x the value of the expression on the right hand side. (eg : x += y, x += « abc »)
<=>	Swap the values of x and y. Where x and y are mutable strings. (eg : x <=> y)
[]	The string size operator returns the size of the string being actioned. (eg : x[] == 3, « abc »[] == 3)

### Control Structures

STRUCTURE	DEFINITION
if	If x is true then return y else return z. (eg : if (x, y, z))
if-else	The if-else statement. Subject to the condition branch, the statement will return either the value of the consequent or the alternative branch. (eg : if (x > y) z ; else w ;)

STRUCTURE	
	DEFINITION
switch	The first true case condition that is encountered will determine the result of the switch. If none of the case conditions hold true, the default action is assumed as the final return value. (eg : switch { case x > 5 : y ; case x < 3 : z ; default : w ; } ;)
while	The structure will repeatedly evaluate the internal statements « while » the condition is true. (eg : while (x > 0) { y += 2 ; x -= 1 ; } ;)
repeat/until	The structure will repeatedly evaluate the internal statements « until » the condition is true. (eg : repeat y += 2 ; x -= 1 ; until x > 0)
for	The structure will repeatedly evaluate the internal statements while the condition is true. On each loop iteration, an « incrementing » expression is evaluated. The conditional is mandatory whereas the initialiser and incrementing expressions are optional. (eg : for (var x := 0 ; x < n ; x += 1) { y += 2 ; } ;)
break	Break terminates the execution of the nearest enclosed loop. (eg : while (y < 10) { if (x < 5) y += 2 ; else break ; } ;)
continue	Continue results in the remaining portion of the nearest enclosing loop body to be skipped. (eg : for (var i := 0 ; i < 10 ; i += 1) { if (i < 5) continue ; j += i ; } ;)
return	Return immediately from within the current expression.
?	Ternary conditional statement, similar to that of the above denoted if-statement. (eg : x ? y : z)
[*]	Evaluate any consequent for which its case statement is true. (eg : [*] { case x > 2 : y += z ; case x > 5 : y += w ; } ;)

### Comments

```
// This is a single line comment
### This is a single line comment
/* This is a multi-line comment */
```

**Vector Processing** Initialisation of a vector can be done in several ways :

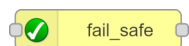
```
var x[3]; # Vector of size 3, all elements are zero
var x[3] := {}; # Vector of size 3, all elements are zero
var x[3] := [ 42 ]; # Vector of size 3, all elements are 42
var x[3] := { 1, 2, 3 }; # Vector of size 3, elements are 1, 2, 3
```

Access to vector elements is done using the following syntax :

```
x[0] := 42; # Assign 42 to the first element of the vector
y := x[0]; # Assign the first element of the vector to y
```

**More** Other examples are available on ExprTk homepage, and more details are available on Readme page.

### 11.33 fail\_safe



#### Description

This Manta box uses several redundant sources and select one to give a safe and robust output. The box can filter out incorrect values. A source can be diagnosed as « invalid » based on several optional criteria : - the validity status of the source is false - the value has been frozen for too long - incorrect values are too frequent (NaN, zero, out-of-range, outliers, etc.) In auto position (see *WebApp controls*), the valid source with highest priority is selected, otherwise, you can manually select the source.

#### Properties

- **Name** : Manta box name
- **Inputs** : Number of sources
- **Ext control** : If selected, add an input to externally control the input to use from the Manta graph.
- **Switch mode** : If selected, FailSafe functionality is disabled and the box behaves like a simple switch.
- **Save** : If selected, current enabled inputs and used input are saved to be restored when Exocet application is restarted.
- **Debug** : If selected, add an output to export useful variables for debug.

- **Failure filter ratio threshold (%)** : It defines the maximum allowable ratio of incorrect values before considering a source as invalid. A value is considered incorrect depending on the checks selected (NaN, zero, empty and/or outlier). The threshold should be set to a level that minimizes false positives (declaring a valid source as invalid) while still capturing an unacceptable ratio of incorrect values. Recommendation :  $threshold [\%] = ( p + 5 \cdot \sqrt{ p \cdot (1-p) / (2 \cdot N - 1) } ) \times 100$ , with p the probability of an incorrect data and N the **Failure filter coefficient**.
- **Failure filter coefficient** : Filtering coefficient N applied to make statistics for each source to determine the ratio of incorrect values. It determines how quickly the ratio is adjusted over time. A higher coefficient results in slower adjustments and makes the ratio determined over a long period. It is comparable to the number of values used to compute the ratio of incorrect values. Recommendation :  $N > (1-p) / 2p$ , with p the probability of an incorrect data.
- **Data freeze time (ms)** : Time (in milliseconds) to consider a source as frozen (see *Check Freeze*).
- **Reject incomplete data streams** : If selected, check if inputs contain all output data.
- **List of variables to rename** (optional) :
  - **Var\_in** : Enter an input variable name (must be unique).
  - **Var\_out** : Enter name to use at the output.
- **List of variables checks** (optional) :
  - **Var\_out** : Enter output name of variable to check (must be unique).
  - **Check NaN** : To filter out NaNs and consider the value incorrect.
  - **Check Zero** : To filter out zeros and consider the value incorrect.
  - **Check Empty** : To filter out empty strings or arrays and consider the value incorrect.
  - **Check Freeze** : If selected, check if value is frozen using *Data freeze time (ms)*. If so, input is considered faulty. Only applied on numbers.
  - **Check OOR** : To filter out out-of-range values and consider the value incorrect. *Minimum* and *Maximum* parameters must be defined below.
  - **Minimum** : Enter minimum limit to *Check OOR* on this variable.
  - **Maximum** : Enter maximum limit to *Check OOR* on this variable.
  - **Check Outlier** : To filter out outlier values and consider the value incorrect.
  - **Max noise** : Enter maximum noise (typical 3 x standard deviation) to *Check Outlier* on this variable.
  - **Max evo** : Enter maximum evolution speed (per second) to *Check Outlier* on this variable.
  - **Check presence** : If selected, check if inputs contain these data.
  - **Validity variable** : If selected, this variable is used to check source validity. Must be a boolean.



- **Invert validity** : If selected, invert boolean logic of this *Validity variable* (default : var=true is OK, inverted : var=false is OK)
  - **Smooth switching** : If selected, smooth an output data gap generated by a position change.
  - **Slope** : If *Smooth switching* selected, enter slope (absolute value per second) to reduce the output data gap.
- **Port labels and timeouts** : For each *Input-x*, enter a timeout (in ms) to consider the input as faulty. 0 is ignored, 1 is minimal value. You can also define a personal label instead of *Input-x*.

## Inputs

**Input\_x** \* Connect your redundant inputs. The input order defines priority : *input-1* is the highest priority.

**Ext control** \* **Select** (*int*) : Define the input to use. If auto is selected (0), input is automatically switched in case of failure detection. \* **Enable\_Input\_x** (*bool*) : Enable or disable input X (enabled by default).

## Outputs

**Valid\_out** \* Main output containing active input variables.

**Status** \* *autoMode* (*bool*) : true if auto position is selected, false otherwise. \* *select* (*int*) : selected input number. \* *selectStr* (*str*) : selected input label. \* *alarm* (*int*) : 1 if problem detected on input and no solution is found, 0 otherwise \* *warning* (*int*) : 1 if problem detected on input and an alternative solution is found, 0 otherwise \* *status* (*int*) : Box status. See *Notes* below. Only enabled inputs are considered. \* *state\_icon* (*uchar array, json*) : Box state icon as JSON object. \* *Input\_X\_status* (*int*) : Input status. See *Notes* below. All inputs are considered. \* *enabledInputListStatus* (*uchar array, json*) : List of enabled input status as JSON array. \* *filteringBadData* (*bool*) : true if the box currently filter out data because an error has been detected and position is not yet changed

**Debug** \* *Input\_X\_checkFlags* (*int*) : bitfield indicating which errors are detected. See *Notes* below. \* *Input\_X\_errRate* (*float*) : failure filter output, indicates the rate of instantaneous errors detected on input data stream.

## WebApp controls

- **Valid\_out** (*int*) : On your dashboard, use a « Slider » widget to define the input to use. If auto is selected (default), input is automatically switched in case of failure detection.

- **Enable\_Input\_x** (*bool*) : On your dashboard, use Button widgets to enable or disable some inputs (enabled by default).

## Notes

Status convention : \* bit 0 : ON/OFF (0=off 1=on) \* bit 1 : Problem detected (0=valid 1=error) \* bit 2 : Problem severity (0=warning 1=error)

### Box status

State	State icon color	Meaning with slider in auto position	Meaning with slider in manual position
0 (0b000)	« dimgray »	Not used	Not used
1 (0b001)	« lime »	OPERATIONAL. First input is valid and selected	OPERATIONAL. Selected input is valid
3 (0b011)	« orange »	WARNING. First input is invalid but one secondary input is valid	Not used
7 (0b111)	« red »	ERROR. All inputs are invalid	ERROR. Selected input is invalid

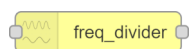
### Input status

State	WebApp color	Meaning
0 (0b000)	« dimgray »	Input off
1 (0b001)	« lime »	Input is valid
3 (0b011)	« orange »	Data received but detected as invalid (future usage)
7 (0b111)	« red »	No data received (input on timeout)

**Check flags**

Bit	Meaning
0x001	NAN value detected
0x002	Zero value detected
0x004	Empty value detected
0x008	Freeze value detected
0x010	Validity check is NOK
0x020	Out Of Range value detected
0x040	Outlier value detected
0x080	Incomplete dictionary
0x100	Presence check is NOK

**Outlier rejection** The failsafe box can reject outliers and consider them as incorrect values if the box **Check Outlier** is checked. The logic to detect an outlier value is based on the following model. The value is considered as a measurement of a parameter. The parameter can evolve with a speed up to **Max evo**. The measurement is considered noisy and is the error of measurement is up to **Max noise**. Based on this model, if the value evolves slowly, it is considered as correct. If the value evolves too rapidly, and neither the parameter evolution speed, nor the measurement noise can explain such a change, the value is considered as incorrect.

**11.34 freq\_divider****Description**

This Manta box performs a sampling frequency division on incoming data.

**Properties**

- **Name** : Manta box name
- **Divider** : Divisor factor to reduce output sampling frequency

- **Outputs** : Number of channels to divide

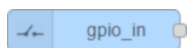
### Inputs

- Connect the data streams containing variables to process. Only one wire must be connected per input.

### Outputs

- Data with divided sampling frequency ( $f_s \rightarrow f_s/\text{divider}$ )

## 11.35 gpio\_in



### Description

This Manta box handles digital input.

### Properties

- **Name** : Manta box name
- **GPIO line** : Select the digital input port.
- **Mode** : In « Edge » mode, return raw edges. In « Frequency » mode, return frequency between rising edges (pulse frequency computation)

Edge mode :

- **Edge** : Select when a signal changes from one state to another.
- **Debouncing time** : When a switch is pressed, it does not provide a clean edge. Adjust the debouncing time to filter out any glitches.
- **Invert** : Invert the digital signal output of the box.
- **Period (s)** : Maximum period between data emission.
- **Add TS** : Add timestamp to output data flow

Frequency mode :

- **Output frequency** : Frequency of output message containing pulse frequency computation
- **Timeout** : If no rising edge is detected during the delay time, the output is set to zero.

## Outputs

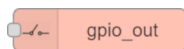
Edge mode :

- Digital input value (*Boolean*, no unit)

Frequency mode :

- EdgeFrequency

### 11.36 gpio\_out



## Description

This Manta box manage digital output.

## Properties

- **Name** : Manta box name
- **GPIO line** : Select the digital output port
- **Type** : Select the output type. It could be STATE (on/off) or PWM (Pulse Width Modulation). The PWM option is only available on port 1. For STATE mode, port 2 & 3 must be preferred.
- **Init state** : Select the initial state of the output.
- **Invert** : Check to invert boolean logic of the output.
- **Frequency** : Enter PWM frequency (from 0.2Hz to 1000Hz). It is the interval of time between successive occurrences of the same state.
- **Duty cycle** : Enter PWM duty cycle (from 0% to 100%). It is the ratio of the high period to the total period of a pulse wave.

## Inputs

- **State** (*Boolean*) : In case of STATE mode, digital signal to generate. In case of PWM mode, permits to disable PWM output.
- **Duty cycle** (*Integer*) : In case of PWM mode, value from 0% to 100% to modify PWM duty cycle.
- **Frequency** (*Float*) : In case of PWM mode, value from 0.2Hz to 1000Hz to modify PWM frequency.

## Notes

Exocet digital outputs are open collector outputs, so active state is near 0 and off state is high impedance. An external pull-up resistor or other electronic circuit is mandatory to raise the output voltage and drive the current when the output is turned off.

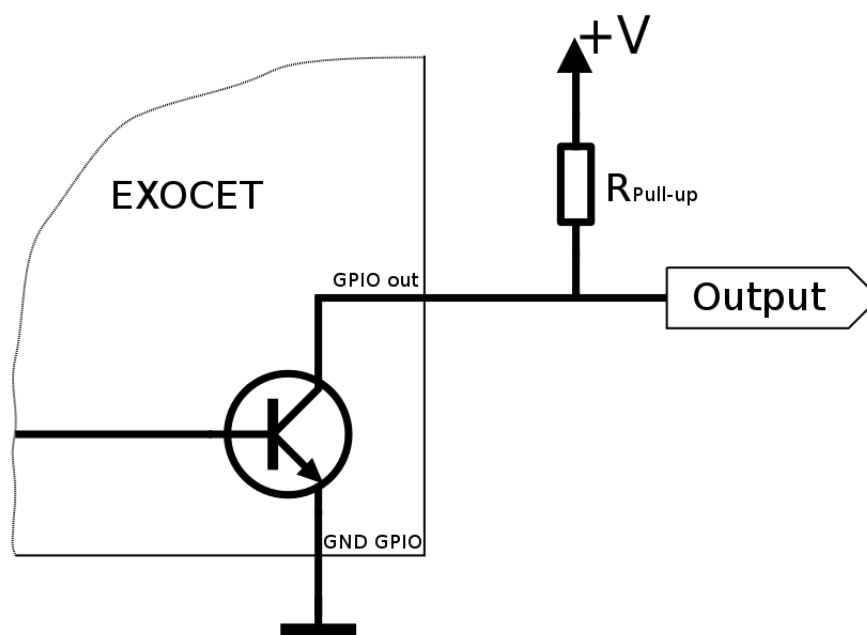


FIG. 81

## 11.37 h5000



### Description

The purpose of this Manta box is to import B&G H5000 sailing instrument variables.

### Properties

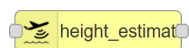
- **Name** : Manta box name
- **Ip** : H5000 IP address
- **Port** : Port number used by H5000 to export data

Use the variable list to check/uncheck the H5000 variables to export. *Select All* and *Unselect All* buttons can be used to select/unselect all the variables.

## Outputs

- The list of selected variables values.

## 11.38 height\_estimation



### Description

Estimates the height and height derivative at specified locations by fusing height and inertial measurements. The box also estimates water elevation and slopes.

The algorithm is a Kalman filter using a model for the inertial navigation system, for the height sensors and for the waves. The wave model is based on the state-of-the-art knowledge on ocean-wave spectra.

The filter is capable to reject false height measurements when they seem unlikely and is thus robust to spray, short-term height sensor failure and even long-term height sensor failure in the presence of height sensor redundancy. ### Properties

#### Box properties

- **Name** Name of the box.
- **Frequency** Output frequency of the box in Hz.

#### INS configuration

- **Inertial sensor gives**

*raw accelerations* : height estimation uses raw acceleration measurements. Raw acceleration measurements do not include gravity, i.e. it is  $\sim 9.81 \text{ m/s}^2$  upward when the sensor is motionless.

*inertial accelerations* : height estimation uses inertial acceleration measurements. Inertial acceleration includes gravity, i.e. it is  $0 \text{ m/s}^2$  when the sensor is motionless.

*heave speed* : height estimation uses heave speed.

- **Get heel and trim accuracy from input** Check this box to provide the  $1\sigma$  accuracy of the pitch and roll angles as inputs of the box.

- **Heel angle 1 $\sigma$  accuracy** Accuracy on the heel angle (~ max. error / 3).
- **Trim angle 1 $\sigma$  accuracy** Accuracy on the trim angle (~ max. error / 3).
- **Gyros ARW** Angular random walk of the gyroscopes (i.e. white noise on the rotation rate measurements).
- **Accel. VRW** Velocity random walk of the accelerometers (i.e. white noise on the acceleration measurements).
- **Accel. bias uncertainty** 1 $\sigma$  accuracy of the uncertain bias on the accelerometers (~ max. error / 3).
- **INS computation point** Coordinates of the point where the inertial navigation unit computes the heave speed or where the accelerations are measured.
- **INS Variable Mapping** Section to define input variable names of the INS data.

#### *Altimeter configuration*

- **Input height min/max** Minimum and maximum of valid height measurements. Measurements outside this range are considered invalid.
- **Pos.** Coordinates of the height sensor.
- **1 $\sigma$  accuracy** Accuracy of the height sensor measurement (~ max. error / 3).
- **Enable update** Check this box to use the sensor for the estimation. Otherwise the validity test is still performed, but the measurement is not used.
- **Height** Input variable name of the height measurement.
- **Timestamp (opt.)** Input variable name of the measurement timestamp. Can be left blank if no timestamp is available.

#### *Estimation options*

- **Swell longitudinal slope** Check this box to estimate the longitudinal slope of the swell. Check this box only if at least two height sensors have different X positions. If unchecked, the longitudinal slope is considered as zero.
- **Swell lateral slope** Check this box to estimate the lateral slope of the swell. Check this box only if at least two height sensors have different Y positions. If unchecked, the lateral slope is considered as zero.  
Check both boxes only if you have at least 3 unaligned height sensors.

#### *Output configurations*

- **Pos.** Coordinates where the height and height derivative is computed.
- **Height** Output height variable name.
- **Height deriv.** Output height derivative variable name.

#### *Advanced options*



- **Boat max. height** Maximum height of the boat with respect to the surface.
- **Max. significant wave height** Maximum significant wave heights that can be encountered.
- **Mahalanobis validity thd** Validity threshold of the  $\chi^2$  test for 1D samples. Default is 6.63.

## Inputs

*Nota : Default units are °, °/s and m.*

Name	Data	Description
<b>INS meas.</b>	<i>Heel angle</i>	Roll angle
	<i>Trim angle</i>	Pitch angle
	<i>Body rot. rate X</i>	Angular speed around the boat x-axis
	<i>Body rot. rate Y</i>	Angular speed around the boat y-axis
	<i>Body rot. rate Z</i>	Angular speed around the boat z-axis
	<i>Heel angle 1<math>\sigma</math> accuracy</i>	Accuracy on the roll angle
	<i>Trim angle 1<math>\sigma</math> accuracy</i>	Accuracy on the pitch angle
	<i>Heave speed</i>	Heave speed (if selected)
	<i>Acc. X</i>	Measured acceleration along the boat x-axis (if selected)
	<i>Acc. Y</i>	Measured acceleration along the boat y-axis (if selected)
<b>Height sensor #i</b>	<i>Acc. Z</i>	Measured acceleration along the boat z-axis (if selected)
	<i>Timestamp</i>	Timestamp of the INS measurement (opt.)
	<i>Height</i>	Height measurement
	<i>Timestamp</i>	Height measurement timestamp (opt.)
	<i>FuseOrder</i>	0 to ignore the measurement (validity test is still done) ; 1 to fuse the measurement only if valid ; 2 to force fusion even if $\chi^2$ deviation is > 1. FuseOrder is also available with a widget on a dashboard.

**Outputs**

Name	Data	Description
<b>Outputs</b>	Timestamp	Timestamp of the output
	<i>Height #i</i>	Height output at location #i (m)
	<i>Height derivative #i</i>	Height derivative output at location #i (m/s)
<b>Status</b>	Timestamp	Timestamp of the status
	Heave	Estimated heave (m)
	Heave_std	Estimated heave 1 $\sigma$ accuracy (m)
	HeaveSpeed	Estimated heave speed (m/s)
	HeaveSpeed_std	Estimated heave speed 1 $\sigma$ accuracy (m/s)
	Height	Height (m)
	Height_std	Height 1 $\sigma$ accuracy (m)
	HeightDerivative	Height derivative (m/s)
	HeightDerivative_std	Height derivative 1 $\sigma$ accuracy (m/s)
	WaveHeave	Wave heave (m)
	WaveHeave_std	Wave heave 1 $\sigma$ accuracy (m)
	WaveHeaveSpeed	Wave heave speed (m/s)
	WaveHeaveSpeed_std	Wave heave speed 1 $\sigma$ accuracy (m/s)
	WaveSlopeX	Wave longitudinal slope (°)
	WaveSlopeX_std	Wave longitudinal slope 1 $\sigma$ accuracy (°)
	WaveSlopeY	Wave lateral slope (°)
	WaveSlopeY_std	Wave lateral slope 1 $\sigma$ accuracy (°)
	AccBias	Vertical bias on measured accelerations (m/s <sup>2</sup> )
	AccBias_std	Vertical bias 1 $\sigma$ accuracy (m/s <sup>2</sup> )

Name	Data	Description
	HeightSensor#i_InvalidRatio   Ratio of invalid measurements	
	HeightSensor#i_Chi2Dev   $\chi^2$ deviation (if >1 the measurement is considered invalid)	

*Nota : Status outputs are expressed at the INS computation location*

## Notes

### Glossary

- **INS** Inertial Navigation System
- **ARW** Angular Random Walk
- **VRW** Velocity Random Walk

**Illustration** Side view of boat

Front view of boat (starboard on the left)

**Reference frame** All coordinates (INS computation location, height sensors and height outputs) must have a common origin. The choice of the origin is at the convenience of the user.

The axis system is SNAME (+x forward/+y starboard/+z downward).

Heave and heave speeds are positive downward.

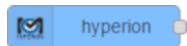
A positive longitudinal wave slope corresponds to an ascent forward. A positive lateral wave slope corresponds to a descent towards starboard.

Height measurements and height outputs are positive upward, i.e. when the sensor or output location is over water.

**Timestamped measurements** Input INS measurement and height measurements can be timestamped. The height estimation can process these measurements chronologically and

interpolates from one to another to be insensitive to data delays. When the input timestamp name is not given (blank field), the algorithm processes the measurement as if it is not delayed.

### 11.39 hyperion



#### Description

Export variables such as microstrains or temperatures from Hyperion, the Micron Optics optical interrogator. It is compatible with si155 and si255 Micron Optics interrogators.

The Hyperion box is configured using a *moi* file. *moi* files are generated by Micron Optics *Enlighth* software. They contain interrogator connectivity parameters, sensors or variables definitions.

**Note :** The grouping notion of Enlighth is not taken into account for now.

#### Properties

- **Name** : Manta box name
- **Conf file** : Select one of the already downloaded *moi* file
- **Fout divider** : Enter a frequency divider to apply on data exported by the Hyperion interrogator
- **Upload new Conf** : Upload a new *moi* file from your PC to the Exocet
- **Delete or Download configuration files** : Use *Trash icon* to delete a configuration file and the *download icon* to download the configuration file from the Exocet to the PC
- **Select variables** : Check/Uncheck the variables to export out of the Manta box
- **Select channel** : Check/Uncheck the channel

#### Inputs

- **Enable** (*bool*) : Connect a boolean to enable/disable the box. Default is enable.

#### Outputs

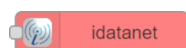
- A set of variables such as microstrains or temperatures according to the configuration file.

## Notes

## Control

- Use the button on the left side of the box to release the access to the interrogator. When released, the Hyperion interrogator can be accessed by *Enlighth Software*.

### 11.40 idatanet



## Description

Export data to iDataNet iPhone/iPad app. Data are sent over UDP to the iOS device.

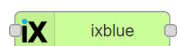
## Properties

- **Name** : Manta box name
- **IP** : IP address of iOS device
- **Port** : UDP port to send data to
- **Decimals** : Number of decimals for exported float numbers
- **Specific decimals** : List of variables that need a specific *Decimals* parameter

## Inputs

- Data to send (multiple wires can be connected).

### 11.41 ixblue



## Description

This Manta box decode several IxBlue Inertial Navigation System Protocols.

**Vessel reference frame** : The data provided by the IxBlue Manta box use the standard SNAME's (1950) notation (forward/starboard/downward and North/East/Down frames).

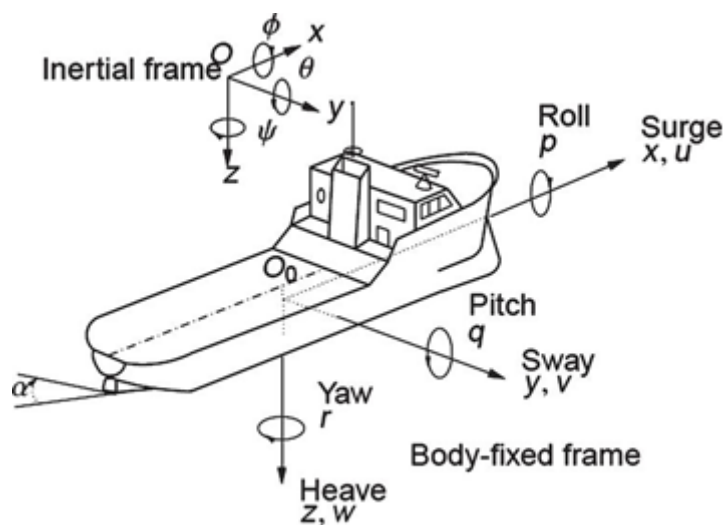


FIG. 82

### Properties

- **Name** : Manta box name
- **Protocol** : Select protocol to decode

### Inputs

- UDP or serial stream to decode.

### Outputs

#### Ifremer Victor :

Data Name	Description
<b>Roll</b>	roll in degrees -180/+180
<b>Pitch</b>	pitch in degrees -180/+180
<b>Heading</b>	heading in degrees 0/360
<b>RollRate</b>	roll rate in degrees per second
<b>PitchRate</b>	pitch rate in degrees per second
<b>YawRate</b>	heading rate in degrees per second

Data Name	Description
<b>SurgeAccel</b>	Surge acceleration in meters per second squared
<b>SwayAccel</b>	sway acceleration in meters per second squared
<b>HeaveAccel</b>	Heave acceleration in meters per second squared
<b>Heave</b>	heave in meters

**INDYN :**

Data Name	Description
<b>Latitude</b>	latitude in degrees -90/+90
<b>Longitude</b>	longitude in degrees -180/+180
<b>Altitude</b>	altitude in meters
<b>Roll</b>	roll in degrees -180/+180
<b>Pitch</b>	pitch in degrees -180/+180
<b>Heading</b>	heading in degrees 0/360
<b>RollRate</b>	roll rate in degrees per second
<b>PitchRate</b>	pitch rate in degrees per second
<b>YawRate</b>	heading rate in degrees per second
<b>SurgeSpeed</b>	surge speed in meters per second

**PIXSE-SUBGP1 :**

Data Name	Description
<b>Time</b>	UTC time
<b>Latitude</b>	in degrees -90/+90, + for north
<b>Longitude</b>	in degrees -180/+180, + for east
<b>RangeToBottom</b>	in meters, if DVL bottom track data is valid
<b>Altitude</b>	altitude in meters
<b>NorthSpeed</b>	in meters per second, + northwards

Data Name	Description
<b>EastSpeed</b>	in meters per second, + eastwards
<b>VerticalSpeed</b>	in meters per second, + upwards
<b>SpeedThWater</b>	in meters per second
<b>Roll</b>	degrees -180/+180, + when port is up
<b>Pitch</b>	degrees -180/+180, + when bow is down
<b>HeadingTrue</b>	in degrees 0/360, 0 is true north
<b>LongVelocity</b>	in meters per second, + forwards
<b>TransVelocity</b>	in meters per second, + to starboard
<b>InsAlgoStatus</b>	INS algo status

**PIXSE-SUBGP2 :**

Data Name	Description
<b>LatitudeStdDev</b>	latitude standard deviation
<b>LongitudeStdDev</b>	longitude standard deviation
<b>RangeToBottomStdDev</b>	range to bottom standard deviation
<b>DepthStdDev</b>	depth standard deviation
<b>NorthSpeedStdDev</b>	north speed standard deviation
<b>EastSpeedStdDev</b>	east speed standard deviation
<b>VerticalSpeedStdDev</b>	vertical speed standard deviation
<b>SpeedThWaterStdDev</b>	SOW standard deviation
<b>HeadingStdDev</b>	heading standard deviation
<b>RollStdDev</b>	roll standard deviation
<b>PitchStdDev</b>	pitch standard deviation
<b>LongVelStdDev</b>	longitudinal velocity standard deviation
<b>TransVelStdDev</b>	transverse velocity standard deviation

**PIXSE-SUBGP3 :**



Data Name	Description
<b>RollRate</b>	in degrees per second, + when increase
<b>PitchRate</b>	in degrees per second, + when increase
<b>YawRate</b>	in degrees per second, + when increase
<b>LongitudinalAcc</b>	in degrees per second squared, + forwards
<b>TransverseAcc</b>	in degrees per second squared, + to starboard
<b>VerticalAcc</b>	in degrees per second squared, + upwards

**CONTROL :**

Data Name	Description
<b>AccXV1</b>	acceleration XV1 in meters per second squared
<b>AccXV2</b>	acceleration XV2 in meters per second squared
<b>AccXV3</b>	acceleration XV3 in meters per second squared
<b>RotRateXV1</b>	rotation rates XV1 in degrees per second
<b>RotRateXV2</b>	rotation rates XV2 in degrees per second
<b>RotRateXV3</b>	rotation rates XV3 in degrees per second

**LONG BIN NAV HR :**

Data Name	Description
<b>Time</b>	data time in seconds
<b>Latitude</b>	latitude in degrees -180/+180
<b>Longitude</b>	longitude in degrees -180/+180
<b>Altitude</b>	altitude in meters
<b>Heave</b>	heave in meters
<b>VelocityNorth</b>	north speed in meters per second
<b>VelocityEast</b>	east speed in meters per second
<b>VelocityDown</b>	down speed in meters per second

Data Name	Description
<b>Roll</b>	roll in degrees -180/+180
<b>Pitch</b>	pitch in degrees -180/+180
<b>Heading</b>	heading in degrees 0/360
<b>RollRate</b>	roll rate in degrees per second
<b>PitchRate</b>	pitch rate in degrees per second
<b>YawRate</b>	yaw rate in degrees per second
<b>UserStatus</b>	INS user status
<b>LatitudeStdDev</b>	latitude standard deviation in meters
<b>LongitudeStdDev</b>	longitude standard deviation in meters
<b>VelocityNorthStdDev</b>	north speed standard deviation in meters per second
<b>VelocityEastStdDev</b>	east speed standard deviation in meters per second
<b>VelocityDownStdDev</b>	down speed standard deviation in meters per second
<b>RollStdDev</b>	roll standard deviation in degrees
<b>PitchStdDev</b>	pitch standard deviation in degrees
<b>HeadingStdDev</b>	heading standard deviation in degrees

**SEAPATH :**

Data Name	Description
<b>Time</b>	data time in seconds
<b>Latitude</b>	latitude in degrees -180/+180
<b>Longitude</b>	longitude in degrees -180/+180
<b>Altitude</b>	altitude in meters
<b>Heave</b>	heave in meters
<b>VelocityNorth</b>	north speed in meters per second
<b>VelocityEast</b>	east speed in meters per second
<b>VelocityDown</b>	down speed in meters per second
<b>Roll</b>	roll in degrees -180/+180

Data Name	Description
<b>Pitch</b>	pitch in degrees -180/+180
<b>Heading</b>	heading in degrees 0/360
<b>RollRate</b>	roll rate in degrees per second
<b>PitchRate</b>	pitch rate in degrees per second
<b>YawRate</b>	yaw rate in degrees per second
<b>Status</b>	status : 0xAA=valid, 0x00=invalid

## 11.42 json\_dec

0000

### Description

Extract data from JSON.

### Properties

- **Name** : Manta box name

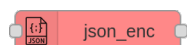
### Inputs

- **Json** (*Opaque*) : Data to decode

### Outputs

- **Data** : Decoded data

## 11.43 json\_enc



## Description

Encode data to JSON.

## Properties

- **Name** : Manta box name
- **Output name** : Exported data name. Manta box name is used if not defined.

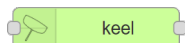
## Inputs

- **Data** : Data to encode

## Outputs

- **Json** (*Opaque*) : Encoded data

## 11.44 keel



## Description

Decode data of IMOCA keel management system.

## Properties

- **Name** : Manta box name
- **System state CAN ID** : Enter CAN ID of System State can frame (hexadecimal)
- **Keel state CAN ID** : Enter CAN ID of Keel State can frame (hexadecimal)
- **Capa state CAN ID** : Enter CAN ID of Capa State can frame (hexadecimal)

## Inputs

- **CAN\_input** : Must be connected to a canbus in box

## Outputs

- **System\_state** : Export general system state data

Nom	Description
<b>RollUp</b>	Compteur de trame continue
<b>NbFsc</b>	Nombre de façade connecté [0..3]
<b>PresenceCapa</b>	0 : capa absente 1 : capa présente
<b>PosSensor</b>	0 : capteur absent 1 : capteur present
<b>StrdPresSensor</b>	0 : capteur absent 1 : capteur present
<b>OVLO</b>	Over Voltage Lock Out : 0 : tension batterie Ok 1 : tension batterie trop importante
<b>UVLO</b>	Under Voltage Lock Out : 0 : tension batterie Ok. 1 : tension batterie trop faible
<b>CCEvRelPort</b>	Court-circuit électrovanne release bâbord : 0 : Ev Ok. 1 : Ev en court-circuit.
<b>CCEvRelStar</b>	Court-circuit électrovanne release tribord : 0 : Ev Ok. 1 : Ev en court-circuit.
<b>CCEvMouvPort</b>	Court-circuit électrovanne mouvement bâbord : 0 : Ev Ok. 1 : Ev en court-circuit.
<b>CCEvMouvStar</b>	Court-circuit électrovanne mouvement tribord : 0 : Ev Ok. 1 : Ev en court-circuit.
<b>CCCmdMotor</b>	Court-circuit commande moteur : 0 : Sortie Ok. 1 : Sortie en court-circuit.

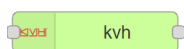
- **Keel\_state** : Export keel state data

Nom	Description
<b>InvPressure</b>	Indique une inversion de poussé de la quille (la quille génère de la portance).0 : la quille génère du couple de redressement (normal).1 : la quille génère de la portance (inversion de poussée). Remarque : De part la géométrie du vérin l'inversion de poussé n'est pas directement lié à une inversion de pression.
<b>KeelAngle</b>	Angle de quille actuel [-38°..38°]. Un angle négatif indique la quille à bâbord, un angle positif indique la quille à tribord.
<b>PortPressure</b>	Pression dans la chambre bâbord (nez) du vérin.
<b>StarboardPressure</b>	Pression dans la chambre tribord (fond) du verin

- **Capa\_state** : Export super capacitors state data

Nom	Description
RollUp	Compteur de trame continue
VBat	Tension batterie
VCapa	Tension supercondensateur
ICapa	Courant des supercondensateurs

## 11.45 kvh



## Description

This Manta box decode some ANPP binary messages from KVH Inertial Navigation System.

Supported messages are : System State Packet (20), Raw Sensors Packet (28), Body Velocity Packet (36), External Body Velocity Packet (47), External Heading Packet (48), Heave Packet (58), North Seeking Initialisation Status Packet (71), Automotive Packet (73)

**Vessel reference frame :** The data provided by the KVH Manta box use the standard SNAME's (1950) notation (forward/starboard/downward and North/East/Down frames).

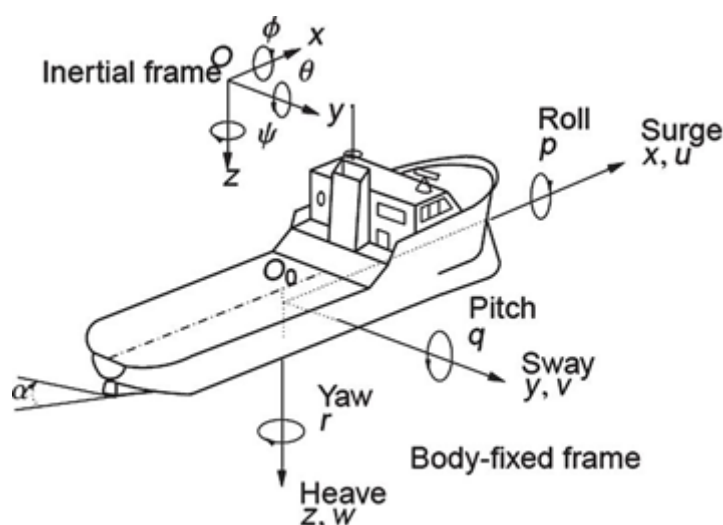


FIG. 83

## Properties

- **Name** : Manta box name
- **Packet ID xx** : If selected, decode this message

## Inputs

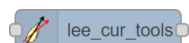
- **Input** : Serial stream to decode.

## Outputs

- **20-System State** : Output data of System State Packet (20)
- **28-Raw Sensors** : Output data of Raw Sensors Packet (28)

- **36-Body Velocity** : Output data of Body Velocity Packet (36)
- **47-External Body Velocity** : Output data of External Body Velocity Packet (47)
- **48-External Heading** : Output data of External Heading Packet (48)
- **58-Heave** : Output data of Heave Packet (58)
- **71-North Seeking Init Status** : Output data of North Seeking Initialisation Status Packet (71)
- **73-Automotive** : Output data of Automotive Packet (73)

## 11.46 lee\_cur\_tools



### Description



FIG. 84

This box realizes calculations in the « SOG, SOW and CUR » Triangle. It contains several independants calculations which are activated on received data and/or optionals properties selected.

Available Calculations are : - Calculate the course and the Speed Over Water (referenced to the course) from the selected sensors. - Get the leeway with current estimation as input. - Estimate the leeway with Hull shape factor formula. - Calculate the boatspeed and leeway using a 2D speedometer sensor. - Estimate the water referenced Boatspeed for a flying boat without speedometer. - Get the current with the leeway estimation as input.

### Properties

- **Name** : Manta box name
- **Calculate Course** : Enable output 0 calculation, see description below
- **Calculate COG-Heading Leeway** : Enable output 1 calculation, see description below
- **Calculate HSF-Leeway** : Enable output 2 calculation, see description below
- **Calculate BoatSpeed** : Enable output 4 calculation, see description below
- **Calculate Current** : Enable output 5 calculation, see description below
- **HullShapeFactor** : Hull Shape Factor ([0..20]), used to calculate output 2



- **Max Leeway** (°) : Limit calculated leeway to +/- this maximum
- **Min SOG** (kn) : Minimum speed over ground, no Leeway output at low speed (inaccurate COG)
- **Use Velocity North East** : Use North East Ground Velocity as input instead of COG, SOG

## Inputs

### Input 0

Default timeout = 2s.

Value Name	Units	Description	Calculations using this input
<b>Leeway</b>	(°)	Leeway angle	- Course, SOW (Output 0) - BSP (Output4) - Current (Output 5)

Note : For a leeway measured with a sensor, you can interface it with a net2000 « leeway » input or the NMEA0183 \$xxLWY sentence.

### Input 1

Default timeout = 1s.

Value Name	Units	Description	Calculations using this input
<b>Heel</b>	(°)	Heel angle	- Hull Shape Factor Leeway calculation (Output 2)
<b>Heading</b>	(°T)	Heading angle	- Leeway (Output 0) or Current (Output 5) - Course (Output 1)

### Input 2

Default timeout = 2s.

Value Name	Units	Description	Calculations using this input
<b>BoatSpeed</b>	(kn)	Longitudinal Boat Speed ; water referenced	- SOW (Output 0) - HSF Leeway calculation (Output 2) - Current (Output 5)

**Input 3** COG, SOG or North East Ground Velocity if *Use Velocity North East* is checked. Default timeout = 2s.

Value Name	Units	Description	Calculations using this input
<b>COG</b>	(°T)	CourseOver Ground	- Current (Output 5) or Leeway (Output 1) - SOW and BSP (Output 4)
<b>SOG</b>	(kn)	Speed Over Ground	- Current (Output 5) or Leeway (Output 1) - SOW and BSP (Output 4)

Value Name	Units	Description	Calculations using this input
<b>VelocityNorth</b>	(m/s) or (kn)	Ground Velocity North	- Current (Output 5) or Leeway (Output 1) - SOW and BSP (Output 4)
<b>VelocityEast</b>	(m/s) or (kn)	Ground Velocity East	- Current (Output 5) or Leeway (Output 1) - SOW and BSP (Output 4)

#### Input 4

Default timeout = 2s.

Value Name	Units	Description	Calculations using this input
<b>LongitudinalSpeedWater</b>	(kn)	Longitudinal Boat Speed, water referenced	- Leeway, BSP and SOW (Output 3)
<b>TransverseSpeedWaterReferenced</b>	(kn)	Transverse Boat Speed, water referenced	- Leeway, BSP and SOW (Output 3)

Note : You can connect this input to a net2000 « boat speed 2D » input, PGN 130578.

#### Input 5

Default timeout = 5s.

Value Name	Units	Description	Calculations using this input
<b>CURD</b>	(°T)	Tidal stream or ocean current direction	- BSP, SOW (Output 4) - Leeway (Output 1)
<b>CURS</b>	(kt)	Tidal stream or ocean current speed	- BSP, SOW (Output 4) - Leeway (Output 1)

Note : You can get a grib current from your navigation software using the NMEA0183 \$xxVDR sentence

## Outputs

### Output 0

Require *Calculate Course* properties to be ticked to be calculated. Calculated at the INS stream rate.

Value Name	Units	Description	Necessary inputs to calculate this data
<b>Leeway</b>	(°)	Leeway angle	Copied from Input 0
<b>Course</b>	(°T)	Course	- Leeway (Input 0) - Heading (Input 1)
<b>SpeedOverWater</b>	(kn)	Speed Over Water (Referenced in Course direction)	- Leeway (Input 0) - BSP (Input 2)

### Output 1

Require *COG-Heading Leeway* properties to be ticked and *MaxLeeway* (°) and *MinSOG* (kt) to be filled to be calculated. Calculated at the GNSS stream rate.

Value Name	Units	Description	Necessary inputs to calculate this data
<b>Leeway</b>	(°)	Leeway angle	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)
<b>BoatSpeed</b>	(kn)	Longitudinal Boat Speed, water referenced	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)

Value Name	Units	Description	Necessary inputs to calculate this data
<b>Course</b>	(°T)	Course	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)
<b>SpeedOverWater</b>	(kn)	Speed Over Water (Referenced in Course direction)	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)
<b>Leeway_orig</b>	(°)	Leeway angle before current correction	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)
<b>BoatSpeed_orig</b>	(kn)	Longitudinal Boat Speed, ground referenced (before current correction)	- Heading (Input 1) - COG, SOG (Input 3) - TideSet, TideRate (Input 5)

## Output 2

Require *HSFLeeway* properties to be ticked and *HullShapeFactor* to be filled to be calculated. Accurate only for a sailboat without moving appendage. Calculated at the INS stream rate.

Value Name	Units	Description	Necessary inputs to calculate this data
<b>Leeway</b>	(°)	Leeway angle estimated using the Hull Shape Factor method	- Heel (Input 1) - BSP (Input 2)

## Output 3

Value Name	Units	Description	Necessary inputs to calculate this data
<b>Leeway</b>	(°)	Leeway angle calculated using 2 axes speedometer	LongitudinalSpeedWaterReferenced, TransverseSpeedWaterReferenced (Input 4)

Value Name	Units	Description	Necessary inputs to calculate this data
<b>BoatSpeed</b>	(kn)	Longitudinal Boat Speed, water referenced	2 axes speedometer (input 4)
<b>SpeedOverWater</b>	(kn)	Speed Over Water (Referenced in Course direction)	2 axes speedometer (input 4)

#### Output 4

Require *CalcBoatspeed* propertie to be ticked to be calculated. Calculated at the INS stream rate.

Value Name	Units	Description	Necessary inputs to calculate this data
<b>BoatSpeed</b>	(kn)	Longitudinal Boat Speed, water referenced,	- Leeway (Input 0) - COG, SOG (input 3) - TideRate, TideSet (Input 5)
<b>SpeedOverWater</b>	(kn)	Speed Over Water (Referenced in Course direction)	- COG, SOG (input 3) - TideRate, TideSet (Input 5)

#### Output 5

Require *CalcCurrent* propertie to be ticked to be calculated. Calculated at the BoatSpeed stream rate.

Value Name	Units	Description	Necessary inputs to calculate this data
<b>CURD</b>	(°T)	Tidal stream or ocean current direction	- Leeway (Input 0) - Heading (Input 1) - Boatspeed (Input 2) - COG, SOG (Input 3)
<b>CURS</b>	(kt)	Tidal stream or ocean current speed	- Leeway (Input 0) - Heading (Input 1) - Boatspeed (Input 2) - COG, SOG (Input 3)

## WebApp controls

- **HullShapeFactor** (float) : If *Calculate HSF-Leeway* is selected, on your dashboard, use a « Set Number » widget to change the *HullShapeFactor* value.
- **MaxLeeway** (float) : If *Calculate COG-Heading Leeway* or *Calculate HSF-Leeway* is selected, on your dashboard, use a « Set Number » widget to change the *MaxLeeway* value.
- **CurrentCorrection** (bool) : If *Calculate COG-Heading Leeway* or *Calculate BoatSpeed* is selected, on your dashboard, use a « button » widget to activate or deactivate current correction.

## Notes

### Leeway, Current, Course definitions :

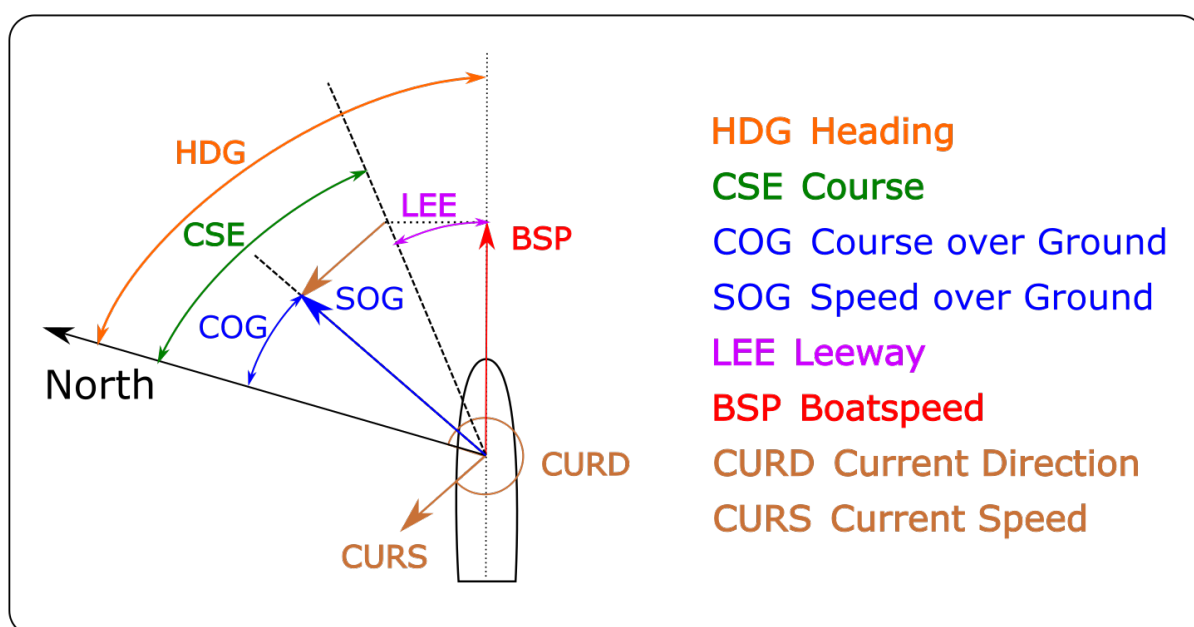
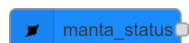


FIG. 85

## 11.47 manta\_status



### Description

This box exports a status of Manta boxes at 5 Hz.

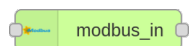
## Properties

- **Name** : Manta box name
- **Out error names** : Select to output names of boxes seen at least one time in error state
- **Out warning names** : Select to output names of boxes seen at least one time in warning state

## Outputs

- **error** (*boolean*) : true if a box is on error, else false
- **error\_nb** (*integer*) : number of boxes on error
- **error\_x** (*string*) : name of boxes on error
- **warning** (*boolean*) : true if a box is on warning, else false
- **warning\_nb** (*integer*) : number of boxes on warning
- **warning\_x** (*string*) : name of boxes on warning

## 11.48 modbus\_in



## Description

This Manta box import data from a remote Modbus device. It works as a master / client, the remote device should be a slave / server.

## Properties

- **Name** : Manta box name
- **Context** : Select Modbus context : RTU or TCP
- **Master** : Check to work as a master. Otherwise, it works as a slave.
- **Serial port** : RTU context only. Serial port configuration box. Press the pencil icon to create a new one. Fill the serial port parameters, then press the *Add* button. Once created, it can be shared with other boxes.
- **Remote IP** : TCP context only. Enter IP address of the remote device to talk.
- **Remote port** : TCP context only. Enter used port of the remote device (default Modbus port is 502).
- **Slave ID** : Define the slave ID of the remote device to talk (from 1 to 247, broadcast=0, tcp=255).
- **List of periodic requests to send** (master mode only, maximum 100) :

- **Function code** : Select the requested function code
- **Address** : Address to start read
- **Number** : Number of data to read (max 100)
- **Period** : Period of request in milliseconds
- **Mapping of data** (slave mode only, maximum 100) :
  - **Type** : Select data type
  - **Address** : Address of data
  - **Variable** : Optional, enter name of variable. If not set, variable name is *Type\_Address*.

## Inputs

### Inputs 0 : Read request

Name	Type	Description
<b>functionCode</b>	Int	1 : read bits (coils) ; 2 : read input bits ; 3 : read registers (Holding) ; 4 : read input registers
<b>address</b>	Int	Address to start read
<b>number</b>	Int	Number of data to read (max 100)

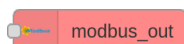
## Outputs

- **data** (*Int*) : read data. Variable names start with *bit\_*, *inputBit\_*, *register\_* or *inputRegister\_*, and are followed by the data address.

## Notes

COM3 to COM6 use an USB extension and are so a little less effective. COM1 and COM2 must be preferred.

### 11.49 modbus\_out





## Description

This Manta box export data to a remote Modbus device. It could work as a master / client, in this case it sends request to write data to a remote device ; or as a slave / server, in this case it make data available in a Modbus object dictionary.

## Properties

- **Name** : Manta box name
- **Context** : Select Modbus context : RTU or TCP
- **Master** : Check to work as a master. Otherwise, it works as a slave.
- **Serial port** : RTU context only. Serial port configuration box. Press the pencil icon to create a new one. Fill the serial port parameters, then press the *Add* button. Once created, it can be shared with other boxes.
- **Remote IP** : TCP context, master mode only. Enter IP address of the remote device to talk.
- **Remote port** : TCP context, master mode only. Enter used port of the remote device (default Modbus port is 502).
- **Local port** : TCP context, slave mode only. Enter used local port (default Modbus port is 502).
- **Slave ID** : Define the slave ID of the remote device to talk in master mode, or define the local slave ID otherwise (from 1 to 247, broadcast=0, tcp=255).
- **Mapping of data** (slave mode only, maximum 100) :
  - **Variable** : Enter name of variable to store (int or bool)
  - **Type** : Select data type
  - **Address** : Address of data

## Inputs

### Inputs 0 : Write request (master mode)

Name	Type	Description
<b>functionCode</b>	Int	5 : write bit (coil) ; 6 : write register ; 15 : write bits (coils) ; 16 : write registers ; 23 : write/read registers
<b>address</b>	Int	Address to write

Name	Type	Description
<b>data</b>	Bool, int, opaque or FloatAr	Data to write. Use opaque to write multiple bits (one byte = one bit). Use floatArray to write multiple registers (one float = one register).
<b>read_addr</b>	Int	Only used if <i>functionCode</i> =23, Address to start read
<b>read_nb</b>	Int	Only used if <i>functionCode</i> =23, Number of data to read (max 100)

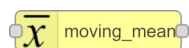
### Inputs 0 : Data to store (slave mode)

See *Mapping of data*

### Notes

COM3 to COM6 use an USB extension and are so a little less effective. COM1 and COM2 must be preferred.

## 11.50 moving\_mean



### Description

This Manta Box can perform average, standard deviation, min/max search and others on a moving set of samples, and on a selected variables list.

### Properties

- **Name** : Manta box name
- **Pass through** : export all non processed data
- **Dynamic window** : add inputs to control windows from the Manta graph
- **WebApp** : Allow window parameters access from Exocet WebApp dashboards
- **Min / Max / Step** : (WebApp option only) widget settings for window parameters

- **Export parameters** : add an output connector to export current parameters
- **Apply to all** : If checked, apply the same process to all variables of the input data stream. Variables are suffix with the process type.
- **Process type** : (Apply to all option only) Select the desired process type (see *Notes* below).
- **Percent** : (Apply to all option only) Only for percentile process type, set the threshold for the percentile process.
- **Window** : (Apply to all option only) Set the period of time to process (in seconds). This period can't be higher than 30min.
- **Auto suffix** : (Apply to all option only) Check to add an automatic suffix to all variables, else use the input field to define a custom suffix.
- **List of variables to process** (maximum 250) :
  - **Process type** : Select the desired process type (see *Notes* below).
  - **Var\_in** : Enter the name of variable to process
  - **Var\_out** : Enter the output name of processed variable
  - **Window** : Set the period of time to process (in seconds). This period can't be higher than 30min.
  - **Percent** : Only for percentile process type, set the threshold for the percentile process.
- **Frequency divider** : Divisor factor to reduce output sampling frequency

## Inputs

- **Data** : Connect here the data stream containing variables to process. Only one wire should be connected on Data input.
- **window** : Connect here a scalar to configure the « Window » dynamically.
- **Reset** : Connect here a boolean to reset dynamic and WebApp parameters to initial values defined above.

## Outputs

- **Processed\_data** : A list of processed data, and non processed data if *Pass through* option is selected.
- **Parameters** : The list of current processing windows.

## WebApp controls

- **Output Name(s)** (*int*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change the *Window* value.

## Notes

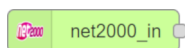
### Max samples :

The max number of samples per box is 30000 (for example : 10 variables at 10Hz for 5min =>  $10 \times 60 \times 5 \times 10 = 30000$  samples).

### Process type :

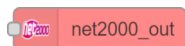
- **Average** : Get the sum of samples divided by the number of samples.
- **Variance** : Return how far the set of samples are spread out from their mean value.
- **Standard deviation** : Return the dispersion of samples from the mean value.
- **Mean deviation** : Return the average dispersion of samples from the mean value.
- **Min** : Get the min value from the current set of samples.
- **Max** : Get the max value from the current set of samples.
- **Median** : Get the middle value from the current set of samples, i.e. the value that divide the set in two equal parts.
- **Percentile** : Get the given percentile value from the current set of samples, i.e. the value below which the given percentage falls (0% gets the minimum, 100% gets the maximum, 50% gets the median).
- **Range** : Get the *Max* minus *Min* value.
- **Max of abs** : Get the max of absolute values of samples.

### 11.51 net2000\_in



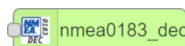
Select a source to see more help

### 11.52 net2000\_out



Select a source to see more help

### 11.53 nmea0183\_dec



## Description

NMEA0183 protocol decoder.

Decoded sentences are : \$BWC, \$GGA, \$GLL, \$HDG, \$HDM, \$HDT, \$INDYN, \$LWY, \$MWV, \$PCBT, \$PENON, \$PHLIN, \$PHSPD, \$PHTRO, \$PIXEL, \$PRDC, \$PRDID, \$PRDM, \$PSBGB, \$PSBGI, \$PSXN, \$PTXT, \$RMB, \$RMC, \$VDR, \$VTG, \$VWR, \$XDR, \$XTE, \$ZDA, \$ZTG, !VDO !VDM.

## Properties

- **Name** : Manta box name
- **Checksum Validation** : If checked, only sentences with a valid checksum are transmitted.
- **NMEA0183 sentences to decode** : Press *New NMEA0183 sentence* button. Select prefix of frames to decode.

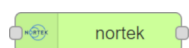
## Inputs

- UDP or serial frame to decode.

## Outputs

- Parsed sentence. One output per selected prefix.

## 11.54 nortek



## Description

This Manta box decode some Nortek DVL binary messages.

Supported messages are : DF3 - Current Profile Data

## Properties

- **Name** : Manta box name
- **Output convention** : Select Raw to have output as defined by Nortek

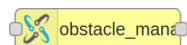
## Inputs

- **Input** : UDP or serial stream to decode.

## Outputs

- **Main Output** : Main single fields of received message
- **Second Output** : Other single fields of received message
- **Velocity** : Raw velocity data as a buffer + 5 first cells decoded velocity (X-Y-Za-Zb)
- **Amplitude** : Amplitude data as a buffer + 5 first cells decoded amplitude
- **Correlation** : Correlation data as a buffer + 5 first cells decoded correlation
- **AltimeterRaw** : Altimeter Raw data as a buffer
- **EchoSounder** : Echo Sounder data as a buffer
- **PercentGood** : Percent Good data as a buffer

## 11.55 obstacleManager



### Description

This Manta box performs data fusion using data coming from the RADAR, the AIS, the OSCAR camera and the boat state.

### Properties

- **Name** : Manta box name

#### *AIS configuration*

- **AIS activated** : If ticked the AIS measurements will be take into account in the data fusion
- **AIS Position noise** : white noise on the position measurement
- **AIS Speed noise** : white noise on the velocity measurements
- **AIS range threshold** : Maximum range from the position of te boat for which the measurments should be considered
- **AIS validation threshold** : Mahalanobis distance threshold for AIS measurement association. If the Mahalanobis distance to an already considered obstacle computed for the measurment is

below this value, the measurement may be associated to the obstacle (unless another one has a shorter distance).

#### *RADAR configuration*

- **RADAR activated** : If ticked the RADAR measurements will be take into account in the data fusion
- **RADAR Position noise** : white noise on the range measurements
- **RADAR bearing noise** : white noise on the bearing measurements
- **RADAR range threshold** : Maximum range for which the measurements hould be considered for fusion
- **RADAR validation threshold** : Same as AIS for RADAR measurements

#### *OSCAR configuration*

- **OSCAR activated** : If ticked the OSCAR measurements will be take into account in the data fusion
- **OSCAR Position noise** : white noise on the range measurements
- **OSCAR bearing noise** : white noise on the bearing measurements
- **OSCAR range threshold** : Maximum range for which the measurements hould be considered for fusion
- **OSCAR validation threshold** : Same as AIS for OSCAR measurements

#### *Prediction configuration*

- **Prediction processing noise** : white noise applied on the position prediction
- **Prediction frequency** : Rate of obstacle position prediction computation

#### *Miscellaneous*

- **Forgetting threshold** : Threshold on the unertainty of obstacle position. If the unceratainty reaches the threshold, the obstacle is considered lost
- **Obstacle default size** : Obstacle size of no information about it is available.
- **Web app** : If ticked, the previous parameters are accessible and can be modified from the dashboard

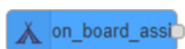
#### **Inputs**

- **OSCAR** : oscar raw measurements coming from the 1st output of the OSCAR manta box.
- **RADAR** : RADAR raw measurements
- **AIS** : AIS measurement.
- **Boat\_State** : Boat state info which is a mix of GNSS and AHRS information. Dictionnary must contain « Latitude », « Longitude, »SOG« , »COG” and « Heading »

## Outputs

- **Main** : Returns info about the obstacles validated one by one
- **Auxilliary** : Returns info about all obstacles validated in one dictionnary
- **Sensors enabled** Returns which sensors info are used in the fusion algorithm

## 11.56 on\_board\_assistant



### Description

Receive events from OnBoardAssistant Sailing performance software, using iDataNet, Adrena or Expedition protocol.

### Properties

- **Name** : Manta box name
- **Port** : Port number to receive data from

## Outputs

### Output 1 : Sails data

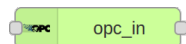
- **Main** (*String*) : main sail description
- **MainReefs** (*String*) : main reef description
- **HeadSail** (*String*) : Head sail description
- **StaySail** (*String*) : Stay sail description
- **Spinnaker** (*String*) : Spinnaker description

### Output 2 : phases data

- **Subject** (*String*) : phase subject
- **StartDate** (*String*) : phase start date
- **Status** (*String*) : phase status
- **Comment** (*String*) : phase comment



## 11.57 opc\_in



### Description

This Manta box import data from a remote OPC-UA device as client.

### Properties

- **Name** : Manta box name
- **URL** : URL of OPC-UA server to connect to
- **Namespace** : OPC-UA *namespaceIndex*
- **Policy** : Select the desired policy. *OnRequest*, periodic read requests are sent. *OnDataChange*, a *subscription* is created on server side with a set of *MonitoredItems*.
- **Vars names** : Select the desired naming convention for output variables. *Custom*, use dedicated field to name Manta variables (*Id* value is used if empty). *Server display names*, use display names defined on server (note : if names are changed on server side, the Exocet operation could be affected).
- **List of data to import (max 100)** :
  - **IdType** : OPC-UA *identifierType*
  - **Id** : OPC-UA *identifier*, that identify a Node in the AddressSpace
  - **Out name** : If *Custom* naming convention, name of output Manta variable
  - **Period** : If *OnRequest* policy, period of request in milliseconds

### Inputs

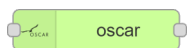
#### Inputs 0 : Read request

Name	Type	Description
<b>IdType</b>	Int	OPC-UA <i>identifierType</i> . 0 : numeric ; 1 : string
<b>Id</b>	Int or String	OPC-UA <i>identifier</i>
<b>VarName</b>	String	Manta variable name to use ( <i>Id</i> value is used if empty or not present)

## Outputs

- **data** : read data

### 11.58 oscar



## Description

This Manta box imports data from Oscar sensor.

## Properties

- **Name** : Manta box name
- **IP** : Enter the IP address of the OSCAR processing unit.
- **Port** : Enter the port of the OSCAR processing unit
- **Detection distance** : objects detected beyond this distance are ignored

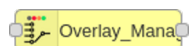
## Outputs

- **Main** : Decoded data, one object at a time
- **Auxiliary** : Decoded data, all objects simultaneously

		Type	Unit	
Data Name				Description
<b>NbObjectsProc</b>	Int			Number of objects detected by OSCAR for which the distance is below the threshold set by the user
<b>MaxDetectionCount</b>	Int			Number of frames for which the object has been tracked continuously
<b>TrackerId</b>	Int			ID of the target
<b>Distance</b>	Float	m		Euclidean distance measured from bow
<b>Angle</b>	Float	°		Bearing of the object from the bow

Data Name	Type	Unit	Description
<b>Size</b>	Float	m	Estimated size of the object
<b>Velocity_x</b>	Float	m/s	Velocity X in SNAME (forward/starboard/downward)
<b>Velocity_y</b>	Float	m/s	Velocity Y in SNAME (forward/starboard/downward)
<b>Velocity_z</b>	Float	m/s	Velocity Z in SNAME (forward/starboard/downward)

## 11.59 overlay\_manager



### Description

This Manta box handles overlay rules cohabitation and communicates the active overlay rule offset to the PilotOverlay box.

Each overlay rule has a number for ID ( $N$ ).

Two different types of overlay rules are managed : **Safety** and **Performance**

#### Safety rules

A safety rule prevents a monitored variable from exceeding a threshold value. Below the threshold, the rule is in standby mode and no action is done. When the threshold is exceeded, an emergency process is triggered. When triggered, it preempts any lower priority safety rule and any active performance rule. Multiple safety rules can run at the same time. Overlay rules with the lowest ID have the highest priority.

When activated a safety rule is in one of these following states : \* Standby : Monitoring variable below the threshold value. The overlay rule can be triggered. \* Triggered (active) : The overlay rule calculates an offset which is sent to the autopilot. \* Preempted : Higher priority safety rule is triggered. The overlay rule can't be triggered.

#### Performance rules

A performance rule modifies autopilot behavior in parallel to normal guidance by applying a positive or negative offset to the user target. Only one performance rule can be active at a given time. When

activated a performance rule is in one of these following states : \* Active : The overlay rule sends offset to the autopilot. \* Preempted : A safety rule is triggered. The overlay rule is paused and automatically switches to active state when the security rule is no longer triggered.

## Properties

- **Name** : Manta box name
- **List of overlay rules to manage** (10 maximum) :
  - **Type** : Overlay rule type (*Safety* or *Performance*)

## Inputs

- **PilotOverlay\_Status** : Must be connected to the « Status » port of the PilotOverlay box.
- **OverlayN\_button** (*Boolean*) : User activation/deactivation of overlay rule *N*. Any variable name is accepted. Can be connected to a « push button » or « gpio\_in » manta box.
- **OverlayN\_Control** : Must be connected to the « Control » port of the overlay rule *N* box.
  - **Offset** (*Float, Degree*) : Overlay rule offset.
  - **RequiredState** (*String*) : Internal state received from overlay rule.

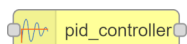
## Outputs

- **PilotOverlay\_Control** : Must be connected to the « Control » port of the PilotOverlay box.
  - **EngagedOverlay** (*Bool*) : To engage/disengage pilot overlay functionality.
  - **Offset** (*Float, Degree*) : Active overlay rule offset.
- **Status** :
  - **EngagedOverlayNumber** (*Integer*) : Number of the current engaged overlay rule. It is the rule that sends offset to the pilot. « 0 » if no overlay rule is engaged.
  - **SafetyTriggered** (*Bool*) : « True » if a Safety rule is triggered, « False » otherwise.
  - **ActivationMsg** (*String*) : To notify that activation failed if activation condition not respected.
  - **OverlayNState** (*String*) : Overlay rule *N* state indicator color.
  - **OverlayNActivationFailed** (*Bool*) : « True » once if the activation of the overlay rule *N* failed.
- **OverlayN\_State** : Must be connected to the « State » port of the overlay rule *N* box.
  - **ManagedState** (*String*) : To control overlay rule state.

## Notes

**Overlay state indicator color convention :** \* **Safety rule** : \* gray = Off \* orange = Activated and preempted by a higher priority safety rule. \* green = Activated in Standby. \* red = Activated and triggered. It is the rule that sends offset to the pilot. \* **Performance rule** : \* gray = Off \* orange = Activated and preempted by a safety rule. \* green = Active. It is the rule that sends offset to the pilot.

## 11.60 pid



### Description

Proportional Integral Derivative controller.

### Properties

- **Name** Name of the box.
- **Frequency** Update frequency of the PID in Hertz.
- **Type** Chosen formulation of the PID. See section **PID Type** below.
- **P,I,D** Check the actions to enable
- **Derivative action** For each of derivative signal (reference and feedback), you can use either an external input (« *explicit input* »), or compute them internally from the reference and feedback using a first order numerical derivation.
- **Filter derivative feedback** Check this to filter the derivative feedback. Only if **Derivative action** is set to *Get derivative feedback from explicit input*. See section **Derivative feedback filter**. It is recommended to use this filter.
- **Derivative filter coefficient N** Ratio between the derivative filter time constant and  $\tau_d$ . Only if **Filter derivative feedback** is enabled. It is recommended to have  $3 < N < 15$ .
- **Coefficients** Coefficients of the PID. See section **PID Type** below.

PID Type	Name	Description
<b>Parallel PID</b>	Kp	Gain applied to the error
	Ki	Gain applied to the integral of the error
	Kd	Gain applied to the derivative error

PID Type	Name	Description
<b>Standard PID</b>	Ks	Gain applied to sum of all actions
	Ti	Inverse of the gain applied to the integral of the error
	Td	Gain applied to the derivative error
<b>Zero-placement PID</b>	Kz	Minimum gain of the PID transfer function when proportional action is enabled.
	$\tau_i$	Time constant of the first zero
	$\tau_d$	Time constant of the second zero

- **Enable deadband** Check this in order to cut-off PID updates when errors are small. See section **Deadband**.
- **Error deadband** Min/max of the deadband on error.
- **Derivative error deadband** Min/max of the deadband on derivative error.
- **Limit output** Check this to saturate the output. Can be disabled only if **Anti-windup method** is not *Clamping*.
- **Output bounds** Min/max of the output.
- **Anti-windup method** Chosen method for anti-windup. See section **Anti-windup**.
- **Autoset back-calculation** Check this to set automatically the back-calculation loop dynamics. See section **Anti-windup**.
- **Back-calculation gain Kb** Gain of the back-calculation loop.
- **Enable feedforward** Check this to input a feedforward signal to add to the PID output. The feedforward is taken into account by the PID to limit the output.
- **Control gains with**
  - *raw values* : control gains with their raw values.
  - *levels* : use an integer between 1 and 9 to control the intensity of the P, I and D actions. See section **Gain control** for more details.
- **Parameters from input** Check this to add an input to control the parameters.
- **WebApp** Check this to control the parameters from the dashboard.
- **Export parameters** Check this to output the parameters.

## Inputs

Name	Data	Description
<b>Reference</b>	<i>Reference</i>	Reference value. Zero is assumed if not fed.
	<i>Derivative reference</i>	Derivative reference value. Zero is assumed if not fed. Only if <b>Derivative action</b> is set to <i>Get derivative reference from explicit input</i> .
<b>Feedback</b>	<i>Feedback</i>	Feedback value.
	<i>Derivative feedback</i>	Derivative feedback value. Only if <b>Derivative action</b> is set to <i>Get derivative feedback from explicit input</i> .
<b>Command feedback</b>	<i>Command feedback</i>	Feedback command from the actuator (e.g. angle of a rudder). The command feedback is used for backcalculation and to initialize the integral term when EnableI is set to true.
<b>Parameters</b>	EnableP	True to enable P action
	EnableI	True to enable I action
	EnableD	True to enable D action
	SetIntegral	True to force integral action to IntegralValue
	IntegralValue	Value used to set the integral action if SetIntegral is true
	Kp	Gain applied to the error (Parallel PID only)
	Ki	Gain applied to the integral of the error (Parallel PID only)
	Kd	Gain applied to the derivative error (Parallel PID only)
	Ks	Gain applied to sum of all actions (Standard PID only)
	Ti	Inverse of the gain applied to the integral of the error (Standard PID only)
	Td	Gain applied to the derivative error (Standard PID only)
	Kz	Minimum gain of the PID transfer function when proportional action is enabled. (Zero-placement PID only)

Name	Data	Description
	Tau <sub>i</sub>	Time constant of the first zero (Zero-placement PID only)
	Tau <sub>d</sub>	Time constant of the second zero (Zero-placement PID only)
	PLevel	Proportional action level
	ILevel	Integral action level
	DLevel	Derivative action level
	N	Ratio between the derivative filter time constant and $\tau_d$
	K <sub>b</sub>	Gain of the back-calculation loop
	ErrMin	Minimum of the error deadband.
	ErrMax	Maximum of the error deadband.
	ErrDotMin	Minimum of the derivative error deadband.
	ErrDotMax	Maximum of the derivative error deadband.
	OutMin	Command minimum.
	OutMax	Command maximum.
<b>Reset</b>	Reset	Reset the parameters to their initial value.
<b>Feedforward</b>	<i>Feedforward</i>	Extra signal added to the PID output. The feedforward is considered to limit the output of the PID.

## Outputs

Name	Data	Description
<b>Command</b>	<i>Command</i>	PID output. Command for the actuator position.
<b>Status</b>	Error	Error
	DerivativeError	Derivative error
	ProportionalAction	Proportional action
	DerivativeAction	Derivative action



Name	Data	Description
	IntegralAction	Integral action
<b>Parameters</b>	EnableP	True to enable P action
	EnableI	True to enable I action
	EnableD	True to enable D action
	StopIntegration	True to stop integration
	Kp	Gain applied to the error (Parallel PID only)
	Ki	Gain applied to the integral of the error (Parallel PID only)
	Kd	Gain applied to the derivative error (Parallel PID only)
	Ks	Gain applied to sum of all actions (Standard PID only)
	Ti	Inverse of the gain applied to the integral of the error (Standard PID only)
	Td	Gain applied to the derivative error (Standard PID only)
	Kz	Minimum gain of the PID transfer function when proportional action is enabled. (Zero-placement PID only)
	TauI	Time constant of the first zero (Zero-placement PID only)
	TauD	Time constant of the second zero (Zero-placement PID only)
	PLevel	Proportional action level
	ILevel	Integral action level
	DLevel	Derivative action level
	N	Ratio between the derivative filter time constant and $\tau_d$
	Kb	Gain of the back-calculation loop
	ErrMin	Minimum of the error deadband.

Name	Data	Description
	ErrMax	Maximum of the error deadband.
	ErrDotMin	Minimum of the derivative error deadband.
	ErrDotMax	Maximum of the derivative error deadband.
	OutMin	Command minimum.
	OutMax	Command maximum.

## Documentation

### PID Type

**Parallel PID** The parallel PID has the following equation :

$$u = K_p \cdot e_p + \int_0^t K_i \cdot e_p \cdot dt + K_d \cdot e_v$$

where :

- $e_p$  is the error
- $e_v$  is the derivative error
- $u$  is the PID output

**Standard PID** The standard PID has the following equation :

$$u = K_s \cdot (e_p + \int_0^t \frac{1}{T_i} \cdot e_p \cdot dt + T_d \cdot e_v)$$

- $T_i$  can be interpreted as the duration required to eliminate the mean of all past errors.
- $T_v$  can be interpreted as the prediction time on future errors.

**Zero-placement PID** The zero-placement PID is defined by the value of its two zeros and its minimum gain. For the P, I, PI, and PD, the equation is as follows, removing the unused terms :

$$u = K_z(e_p + \int_0^t \frac{1}{\tau_i} \cdot e_p \cdot dt + \tau_d \cdot e_v)$$

For the PID, the equation is :

$$u = K_z \left( \frac{\tau_i + \tau_d}{\tau_i} e_p + \int_0^t \frac{1}{\tau_i} \cdot e_p \cdot dt + \tau_d \cdot e_v \right)$$

The corresponding frequency response is :

*Zero-placement PID frequency response*

- Errors having a frequency below  $1/\tau_i$  rad/s are eliminated by the integral action.
- Errors having a frequency over  $1/\tau_d$  rad/s are anticipated.

**Deadband** The deadband is defined as the intersection of a error between [pmin, pmax] and a derivative error between [vmin,vmax]. The control is disabled in the deadband, thus when the error and the derivative error **are both** in the deadband intervals. The deadband on derivative error is enabled only if the derivative action is enabled. The control output is continuous when entering and leaving the deadband.

**Anti-windup** If the PID output is bounded using **Limit output**, when the controller reaches the actuator limits, the integral term accumulates a significant error and becomes really large. This is called windup. A controller windup can cause a significant delay when the error changes its sign before the PID output leaves its bound. Using the anti-windup prevents the controller to windup. Two methods are proposed :

**Clamping** The clamping anti-windup method consists in stopping the integration of the error when the output has reached a bound. This option can be selectable only if **Limit output** is selected. This method is recommended when only an internal saturation is used.

**Back-calculation** The back-calculation method consists in unwinding the integral of the error when the output has reached a bound. The following block-diagram illustrates the back-calculation loop.

*Back-calculation loop with external input*

*Back-calculation loop with internal input*

Back-calculation gain can be automatically set using the option **Autoset back-calculation**.

Back-calculation can be convenient for dynamic systems with large delays. Back-calculation can also be used to track the actuator target and therefore ensure a bumpless controller switch.

**Gain control** Gain control can be done with levels using the option **Control gains with levels**. When this options is set. The input or dashboard variables PLevel, ILevel and DLevel can be set between 1 and 9. A coefficient corresponds to each level :

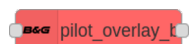
Level	1	2	3	4	5	6	7	8	9
Coefficient	0.41	0.51	0.64	0.80	1.00	1.25	1.56	1.95	2.44

A level increment increases the corresponding action (P, I or D) by 25%. These coefficients impact the following variables :

	PLevel +1	ILevel +1	DLevel +1
Parallel PID	Kp +25%	Ki +25%	Kd +25%
Standard PID	Ks +25%	Ti -25%	Td +25%
Zero-placement PID	Kz +25%	$\tau_i$ -25%	$\tau_d$ +25%

**Input timeout management** If the **Feedback** input or **Actuator position** input have timed out, the PID controller stops updating and its state switches to *WARNING*. Once the timed-out input is eventually refreshed, the PID resumes with no discontinuity.

## 11.61 pilot\_overlay\_bandg



### Description

Ease of use B&G H5000 PILOT interface :

- Virtual Pilot Controller (on dashboard with button widgets)
- Pilot overlay for Safety or performance purposes :
  - Add an external « Offset » to the pilot target (WindReference or HeadingToSteer), set by the skipper :  $CorrectedTarget = UserTarget + Offset$
  - Works in « TrueWind » mode and « Heading » mode. « ApparentWind », « No Drift » and « Navigation » modes are not taking into account. This box must not be used with an H5000 PILOT in one of these modes.

- Disengages automatically when H5000 PILOT disengage, change mode, port/starboard 10° order, tack/gybe order and if lost communication with the autopilot.
- Disengages on « EngageOverlay » falling edges.
- When disengage, the H5000 PILOT keeps its last target value (« FeedbackTarget »)
- Engages on « EngageOverlay » rising edges if engagement conditions are respected.
- Engagement conditions :
  - \* H5000 PILOT connected, in TrueWind or Heading mode
  - \* « Offset » received

## Properties

- **Name** : Manta box name
- **Offset** : Mapping name of the offset
- **EngageOverlay** : Mapping name of the engage control signal
- **Offset MaxVariation** : If the variation between two successive Offset is above, ignore the last Offset value.
- **UserTarget MaxVariation** : If the variation between two successive UserTarget is above, ignore the last UserTarget value.

## Inputs

- **Control** : Pilot Overlay control inputs

Variable	Type	Unit	Description
<b>Offset</b>	Float	Degrees	Heading or TrueWind angle offset
<b>EngageOverlay</b>	Bool		Engage/Disengage Pilot Overlay functionality

- **65305-SIMRAD Modes/Heartbeat** : Must be connected to a « Pilot Controller net2000\_in » Box with « Address Filter » to « All »
- **130850-SIMRAD Event Command** : Must be connected to a « Pilot Controller net2000\_in » Box with « Address Filter » to « All »
- **65341-SIMRAD Autopilot Sailing Mode** : Must be connected to an « Autopilot net2000\_in » Box with « Address Filter » to « All »
- **127237-Heading Track Control** : Must be connected to an « Autopilot net2000\_in » Box with « Address Filter » to « All »

- **Button <10** : May be connected to a button box (periodicity = None)
- **Button 10>** : May be connected to a button box (periodicity = None)
- **Button <1** : May be connected to a button box (periodicity = None)
- **Button 1>** : May be connected to a button box (periodicity = None)
- **Button STBY** : May be connected to a button box (periodicity = None)
- **Button AUTO** : May be connected to a button box (periodicity = None)
- **Button MODE** : May be connected to a button box (periodicity = None)

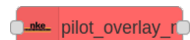
## Outputs

- **65305-SIMRAD Modes** : Must be connected to a PilotControllerOut Box with « Address Filter » to « All »
- **130845-SIMRAD Parameter Handle** : Must be connected to a PilotControllerOut Box with « Address Filter » to « All »
- **130850-SIMRAD Event Mode** : Must be connected to a PilotControllerOut Box with « Address Filter » to « All »
- **130850-SIMRAD Event ChangeRef** : Must be connected to a PilotControllerOut Box with « Address Filter » to « All »
- **Status** :

Variable	Type	Unit	Description
<b>PilotEngaged</b>	Bool		« True » / « False »
<b>Mode</b>	Text		« Stop », « Heading », « TrueWind »
<b>PilotMode</b>	Int		0 = « Stop », 1 = « Heading », 2 = TrueWind”
<b>PilotSuperModeOn</b>	Bool		Always « False », because B&G H5000 Pilot doesn't have super modes.
<b>UserTarget</b>	Float	Degrees	Target set by the skipper. Can be change by +/-1° when pilot overlay engaged
<b>CorrectedTarget</b>	Float	Degrees	« UserTarget » + « Offset »
<b>FeedbackTarget</b>	Float	Degrees	Actual H5000 PILOT TrueWind or Heading target
<b>OverlayEngaged</b>	Bool		« True » / « False »

Variable	Type	Unit	Description
<b>OverlayDisengageEvent</b>	Text		« None », « TackGybe », « Port10 », « Starboard10 », « ChangeMode », « PilotDisengage », « EngageOverlayInput », « PilotComTimeout »
<b>PilotComTimeout</b>	Bool		« True » if one timeout on the input ports coming from « Autopilot In » box is triggered
<b>InvalidOffsetCount</b>	Int		Number of Offset discarded due to too large variation
<b>InvalidWindTargetCount</b>	Int		Number of WindTarget discarded due to too large variation
<b>InvalidHeadingTargetCount</b>	Int		Number of HeadingTarget discarded due to too large variation

## 11.62 pilot\_overlay\_nke



### Description

Ease of use NKE PILOT interface :

- Virtual Pilot Controller (on dashboard with button widgets)
- Pilot overlay for safety or performance purposes :
  - Add an external « Offset » to the pilot target (Wind or Heading), set by the skipper :  $CorrectedTarget = UserTarget + Offset$
  - Works in « TrueWind », « ApparentWind », and « Heading » mode. « Polar » and « Rudder » modes are not taking into account. This box must not be used with a PILOT in one of these modes.
  - Disengages automatically when PILOT disengage, change mode, active super mode order, port/starboard 10° order and tack/gybe order.
  - Disengages on « EngageOverlay » falling edges.
  - When disengage, the PILOT keeps its last target value (« FeedbackTarget »)
  - Engages on « EngageOverlay » rising edges if engagement conditions are respected.
  - Engagement conditions :

- \* PILOT connected, in TrueWind, ApparentWind or Heading mode
- \* « Offset » received
- \* NKE « Super modes » Off

## Properties

- **Name** : Manta box name
- **Offset** : Mapping name of the offset
- **EngageOverlay** : Mapping name of the engage control signal
- **Offset MaxVariation** : If the variation between two successive Offset is above, ignore the last Offset value.
- **UserTarget MaxVariation** : If the variation between two successive UserTarget is above, ignore the last UserTarget value.

## Inputs

- **Control** : Pilot Overlay rules inputs (must be sent periodically)

Variable	Type	Unit	Description
<b>Offset</b>	Float	Degrees	Heading or Wind angle offset
<b>EngageOverlay</b>	Bool		Engage/Disengage Pilot Overlay functionality

- **SailNet** : Must be connected to a SailNet\_in box correctly configured on Processor HR. Waiting for variables *PilotRef*, *ModePilHR* and *StatusPilHR*.
- **Feedback** : Must be connected to a udp\_in box correctly configured to receive NKE Ethernet Box.
- **Button <10** : May be connected to a button box (periodicity = None)
- **Button 10>** : May be connected to a button box (periodicity = None)
- **Button <1** : May be connected to a button box (periodicity = None)
- **Button 1>** : May be connected to a button box (periodicity = None)
- **Button STOP** : May be connected to a button box (periodicity = None)
- **Button AUTO** : May be connected to a button box (periodicity = None)
- **MODE** : Pilot mode control input for update from the Manta graph (external control)



**Outputs :**

- **Serial out** : Must be connected to a udp\_out box correctly configured to send to NKE Ethernet Box.
- **Status** :

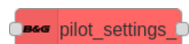
Variable	Type	Unit	Description
<b>PilotEngaged</b>	Bool		« True » / « False »
<b>Mode</b>	Text		« Stop », « Heading », « TrueWind », « ApparentWind »
<b>PilotMode</b>	Int		1 = « Heading », 2 = TrueWind«, 0 = pilot no engaged or other pilotMode    <b>PilotSuperModeOn</b>   Bool    »True” if at least on NKE pilot super mode is on. « False » otherwise.
<b>UserTarget</b>	Float	Degrees	Target set by the skipper. Can be change by +/-1° when pilot overlay engaged
<b>CorrectedTarget</b>	Float	Degrees	« UserTarget » + « Offset »
<b>FeedbackTarget</b>	Float	Degrees	Actual PILOT target (Wind or Heading)
<b>OverlayEngaged</b>	Bool		« True » / « False »
<b>OverlayDisengageEvent</b>	Text		« None », « TackGybe », « Port10 », « Starboard10 », « ChangeMode », « PilotDisengage », « EngageOverlayInput », « PilotSuperModeOn »
<b>PilotComTimeout</b>	Bool		Always « false ». No communication timeout detection with NKE PILOT
<b>InvalidOffsetCount</b>	Int		Number of Offset discarded due to too large variation
<b>InvalidUserTargetCount</b>	Int		Number of UserTarget discarded due to too large variation

- **Super Modes** : State of the NKE pilot « Super modes » : Boost, Gust, PilotTable, Surf.

## WebApp controls

- **PilotModeNKE** (*int*) : On your dashboard, use a « Slider » widget to manually set the pilot mode. It is synchronised with the NKE Processor HR pilot mode.

### 11.63 pilot\_settings\_bandg



## Description

Ease of use B&G H5000 PILOT interface

## Properties

- **Name** : Manta box name

## Inputs

- **PGN 127237** : Must be connected to port 127237 of a net2000 in box with type set as Autopilot
- **PGN 130845** : Must be connected to port 130845 of a net2000 in box with type set as Autopilot
- **Ext control** : Pilot Settings control input for update from the Manta graph

Variable	Type	Unit	Comment
<b>RudderGain</b>	Float		
<b>RudderLimit</b>	Float	deg	
<b>CounterRudder</b>	Float	s	
<b>CruisingSpeed</b>	Float	kn	
<b>ManualSpeed</b>	Float	kn	
<b>BoatLength</b>	Float	m	
<b>PerformanceMode</b>	Int		
<b>AutoTrim</b>	Float	s	
<b>TackAngle</b>	Float	deg	

Variable	Type	Unit	Comment
<b>TackTime</b>	Float	s	
<b>WindMode</b>	Int		1=Apparent, 2=True, 4=Auto
<b>HeelCompensation</b>	Bool		
<b>GustResponse</b>	Bool		
<b>TwrsResponse</b>	Bool		
<b>SteeringAutomaticResponse</b>	Int		0=Off, 1=Economy, 2=Normal, 3=Sport
<b>SteeringRecovery</b>	Int		0=Off, 1=Narrow, 2=Medium, 3=Wide
<b>Adapt</b>	Bool		
<b>OffCourse</b>	Float	deg	
<b>LowBoatSpeed</b>	Float	kn	
<b>RudToRateRatio</b>	Float		
<b>RudderFeedback</b>	Int		1=InternalFrequency, 4=External
<b>DriveVoltage</b>	Float	V	
<b>MotorOutput</b>	Int	%	
<b>ManualDeadband</b>	Float	deg	

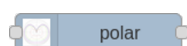
## Outputs

- **STATUS** : Export current Pilot settings
- **Other outputs** : Must be connected to port 130845 of a net2000 out box with type set as Autopilot

## WebApp controls

On your dashboard, use « Slider », « Set number » or « Button » widgets to change parameter values.

### 11.64 polar



## Description

Use polar table to calculate some performance data.

## Properties

- **Name** : Manta box name
- **Ext control** : If checked, add inputs to dynamically manage the performance table
- **Save** : If checked, save dynamic update on the file. It is strongly recommended to not periodically send a dynamic table if this option is enabled.
- **Interpolation** : Select desired interpolation method
- **File format** : Select used file format
- **List of polar tables** :
  - **Name** : Enter a table name
  - **File** : Select a performance polar table (format, see note below)

## Inputs

### Input 0 : True Wind

Name	Units	Description
<b>TWA</b>	°T	True wind angle
<b>TWS</b>	kn	True wind speed

### Input 1 : Boat speed

Name	Units	Description
<b>BoatSpeed</b>	kn	Longitudinal boat speed, water referenced

### Input 2 : Ext polar table

Name	Type	Description
<b>PolarTable</b>	FloatTable	Dynamic polar table. Variable name is free.

**Input 3 : Ext table selection**

Name	Type	Description
<b>TableSelector</b>	Scalar/String	Dynamic table selector. Variable name is free.

**Outputs****Output 0 : Performances**

Name	Units	Description
<b>VMG</b>	kn	Velocity Made Good (VMG) is the component of Boat Speed in the direction of the True Wind.
<b>BSP_Polar</b>	kn	Expected BoatSpeed from the polar table, at current TWA and TWS.
<b>BSP_PolarRatio</b>	%	Ratio of current BoatSpeed on BSP_Polar.
<b>VMG_Polar</b>	kn	Optimum VMG of the polar table, for current TWS and current TWA sector.
<b>VMG_PolarRatio</b>	%	Ratio of current VMG on VMG_Polar.
<b>TWA_PolarTgVmg</b>	°(+/-180)	TWA target to get the optimum VMG of the polar table, for current TWS and current TWA sector.
<b>BSP_PolarTgVmg</b>	kn	BoatSpeed target to get the optimum VMG of the polar table, for current TWS and current TWA sector.

**Notes****VMG :**

Velocity Made Good is defined as the projection of boat speed on the wind axis.

**Tables requirements :**

- B&G WTP/Deckman, B&G H5000 or NKE Processor Regatta compatible format
- TWA must be > 0° and <= 180°
- TWS must be >= 0kn and <= 100kn
- BSP must be >= 0kn and <= 100kn

**Calculations above boundaries :**

- The spline surface interpolation is constrained in order to have a null boat speed at TWA=0°
- The spline surface interpolation is constrained in order to continue with the same boat speed at TWA=180°
- For TWS below the first TWS : return values corresponding to the first TWS
- For TWS above the last TWS : return values corresponding to the last TWS

**Spline surface limitations :**

- The 3D spline interpolation can give bumps if 2 different points are too close and if the TWS/TWA are poorly distributed.
- The spline surface is smoother with only few points per TWS.

**B&G WTP/Deckman file format**

Size : maximum 16 pairs of BSP/TWA, maximum 20 TWS lines.

```

v1  a1  v2  a2  v3  a3  v4  a4  v5  a5
TWS BSP TWA BSP TWA BSP TWA BSP TWA BSP TWA
TWS BSP TWA BSP TWA BSP TWA BSP TWA BSP TWA

```

**B&G H5000 file format**

Size : 17 TWA columns, maximum 20 TWS lines.

```

TWS\TWA 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
TWS    BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP
TWS    BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP

```

**NKE Processor Regatta file format**

Size : 10 TWS columns, 14 TWA lines.

```

TWS TWS TWS TWS TWS TWS TWS TWS TWS TWS
TWA BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP
TWA BSP BSP BSP BSP BSP BSP BSP BSP BSP BSP

```

**Expedition / SailingPerformance file format**

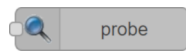
Size : maximum 16 pairs of TWA/BSP, maximum 20 TWS lines.

```

TWA BSP TWA BSP TWA BSP TWA BSP TWA BSP
TWS TWA BSP TWA BSP TWA BSP TWA BSP TWA BSP
TWS TWA BSP TWA BSP TWA BSP TWA BSP TWA BSP

```

## 11.65 probe



### Description

A probe displays data exported by boxes linked with it.

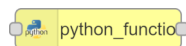
### Properties

- **Name** : Manta box name
- **Full speed** : If checked, every data value changes will be reported. If unchecked, data is refreshed every 300 ms. **IMPORTANT NOTE** : Using **Full speed** can impact Exocet CPU performances if data to report changes frequently. It is recommended to use this option only during setup/debug phase and not in production.
- **History** : The « Show/Hide History » button is used to switch between history/current modes. When in history mode, the most recent data is displayed first. Only 1000 data values are displayed to avoid browser performance issues.
- **Pause** : The « pause/play » button is used to stop refreshing data values.
- **Data download** : In pause mode, two buttons are available to download displayed data as JSON or CSV format file. For CSV, data separator is space.

### Inputs

- Data stream to monitor. Multiple wires can be connected.

## 11.66 python\_function



### Description

A function block where you can write python code to do interesting things.

- Python 2.7 code is supported and the NumPy library as well.
- The Python code is run each time a data is received by an input connector. The *Tick* Manta box can be used to periodically execute the Python code.

- Each Box holds its own local context. This local context is preserved accross each script invocation.
- Python variables can be shared accross boxes by using the *global* keyword, but its use is strongly discouraged since all python boxes of Manta could be impacted.
- To use an external variable inside a function, prefer to use parameters

## Inputs

- Multiple inputs can be defined by using the *Number of Inputs* widget at the bottom.
- The incoming data is accessible from the *msg* Python object. *msg* is a dictionary with variable names as keys.
- The *port* Python variable holds the input connector index on which the data arrives. It is used to know which input connector has triggered the Python script.

## Outputs

- Multiple outputs can be defined by using the *Number of Outputs* widget at the bottom.
- The Python script must define a *rep* Python array to export data out of the box. The *rep* array size must match the number of output connectors.
- Each *rep* element is either a dictionary, either *None*. Each element represents the set of data to export to a specific output connector on script completion. If an element is a dictionary, variables will be exported on the connector with same inded as the element. If an element is *None*, no data will be exported on the corresponding output connector.
- Depending on the Python variable type, *Outputs* data type can be : Boolean, Integer, Float, String, CanFrame, Opaque (max size 2048), FloatArray (max size 1024) or FloatTable (max size 32 x 32).

## Notes

### Intialize a variable at first script invocation

In order to initialize or keep a variable value between all the script invocations, you can use that :

```
try :  
    X  
except NameError :  
    X = 0.0  
...
```



### Hello World exemple

In this exemple, a data named « hello » with String value « Hello World ! » is sent to output connector 0 at each script invocation ;

```
out={}
out["hello"] = "Hello World!"
rep = [ out ]
```

### Handling Input connectors

In this exemple :

- The Python box is assumed to be configured with several input connectors and a single output.
- If script is triggered by input connector 0, incoming data are forwarded to output connector.
- If script is triggered by another input connector than 0, no data is exported.

```
if port==0 :
    out = dict(msg)
else :
    out = None
```

```
rep = [ out ]
```

### Handling Input data

In the example below :

- The function block is assumed to be configured with 2 output connectors.
- The incoming data *msg* is assumed to have three fields : *TWA*, *TWS*, and *BSP*.
- The *rep* variable will output the *out1* object to connector 1, and the *out2* object to connector 2. Thus connector 2 will pass a variable with a single *Debug* field, whereas connector 1 will have the newly calculated *TWS* and *BSP*.

*###reads incoming data fields of msg*

```
trueWindAngle = msg["TWA"]
trueWindSpeed = msg["TWS"]
boatSpeed = msg["BSP"]
```

```
apparentWindAngle = msg["AWA"]
apparentWindSpeed = msg["AWS"]
```

```

out1={}
out2={}

bsp = math.sin(apparentWindAngle)*apparentWindSpeed

out1["TWS"] = math.cos(trueWindAngle)*0.6*trueWindSpeed
out1["BSP"] = bsp

if bsp > 35 :
    out2["Debug"] = "Too fast"
elif (bsp < 1.0) :
    out2["Debug"] = "Too slow"
else :
    out2["Debug"] = "Cool"

rep = [ out1 , out2 ]

```

### Handling Units

A data can be published with a unit. The data must be encapsulated in a list with its unit in a string format (upper case) at first position.

```

out["MyConsumptionMessage"] = ["WATT", 1.3]

rep = [ out ]

```

Current available units are: UNSET, AMPEREHOURS, AMPERES, ANGULAR\_DEGREES, ANGULAR\_DEGREES\_180, ANGULAR\_DEGREES\_360, ANGULAR\_DEGREES\_PER\_SECOND, ANGULAR\_MAGNETIC\_DEGREES, ANGULAR\_TRUE\_DEGREES, BAR, BITFIELD, CANDELA, CELSIUS, CUBIC\_METERS, DATE, DECIBELS, FARENHEIT, FEET, GRAVITY\_ACCELERATION, HERTZ, HOURS, HPA, HUMIDITY, JOULE, KELVIN, KILOBYTES, KILOBYTES\_PER\_SECOND, KILOGRAMS, KILOGRAMS\_PER\_CUBIC\_METER, KILOMETERS\_PER\_HOUR, KILOWATTS, LITERS, LITERS\_PER\_SECOND, MAGNETICFIELD, MEGABYTES, MEGABYTES\_PER\_SECOND, METERS, METERS\_PER\_SECOND, METERS\_PER\_SQUARE\_SECOND, MICROSTRAIN, MILLIAMPERES, MILLIBAR, MILLIMETERS, MILLISECONDS, MILLIVOLTS, MILLIVOLTS\_PER\_VOLT, MILLIWATTS, MINUTES, MOLE, NANOMETERS, NAUTICAL\_KNOTS, NAUTICAL\_MILES, NEWTON, NEWTONMETERS, PASCAL, PERCENT, PPM, RADIANS, RADIANS\_PER\_SECOND, RPM, SALINITY, SECONDS, TESLA, TIME, TONNES, VOLTS, WATT.

### Handling ByteArray

bytearrays are a kind of data, that can come from *UDP*, *Serial*, or other Python box. Bytearray data

outgoing from the Python Box is converted to the Opaque data type.

```
out = {}
```

```
s = "Very interesting!"
b=bytearray(s)
out["data"] = b
rep = [ out ]
```

That leads to an output message containing the string « Very interesting » as an array.

### Display HTML or icons in dashboards

Output data must be in JSON format, converted to the Opaque data type. JSON keys are : \* type : *icon* or *text* \* icon : string containing icons class names. Font-Awesome is used. The first fa class is added automatically. Default-icon is fa-circle. \* text : HTML text \* color (optional) : color of text or icon. Color can be HTML color name or HEX value with # prefix. See Web colors.

```
out = {}
```

```
value = msg['Scalar']
icon = 'fa-circle'
if value == 0 :
    color = '#c0c0c0'
    text = '<h3>Stopped</h3>'
elif value < 7 :
    color = 'green'
    text = '<h3>Running</h3>'
elif value < 9 :
    color = 'orange'
    text = '<h2>Warning</h2>'
else :
    color = 'red'
    icon = 'fa-times-circle'
    text = '<h1>STOP!</h1>'
```

```
### export values in byte arrays
```

```
out["icon"] = bytearray('{"type" : "icon", "icon" : "' + icon + ' fa-2x", "color" : "'
out["text"] = bytearray('{"type" : "text", "text" : "' + text + '"}')

rep = [ out ]
```

Outputs are automatically displayed as icons or HTML in dashboard data widgets.

### Handling CAN Frames

When a CAN box is connected to an input of the python box, the incoming `msg[« can »]` is special. It is of `pixel.CanFrame` type

The `pixel.CanFrame` class has 2 attributes :

- `id` : the CAN Id
- `bytes` : a `ByteArray` of the CAN data (up to 8 bytes)

*Example :*

In this example, regardless of the incoming CAN frame ID, the 2 first bytes of the frame are modified, and frame with ID « 0x457 » is emitted.

```
### Handle input message
canmsg = msg["can"]

###Is CAN frame?
try :
    if type(canmsg) is not pixel.CanFrame :
        raise TypeError('no canframe')
    isCAN = True
except :
    isCAN = False

###Process
if isCAN :
    expectedId = 0x08FF0DFC
    # Test on 29 bits identifier (can 2.0b)
    if ( (canmsg.id & 0x1FFFFFFF) == expectedId ) :
        MSG = pixel.CanFrame(canmsg.id, canmsg.bytes)
    else :
        MSG = pixel.CanFrame(0x08FFFF00, bytearray([0x00, 0x00]))
else :
    MSG = None

### emits the result on the output connector
rep = [ MSG ]
```

### Playing with bytearrays and deque FIFO

In the following example, all incoming bytes are first stored in a deque FIFO (First In, First Out) buffer, and are post-processed later. Notice that the deque buffer is declared as a persistent variable.

The data is added to the deque at the right end through *append* and popped from the left end through *popleft*.

```
from collections import deque
```

```
try :
    buf
    height
except NameError :
    buf = deque()
    height = 0.0

def isBeginOfFrame(b) :
    if ((b & 0x80) == 0x00 and (b & 0x7f) == 0x48) :
        return True
    return False
```

```
### First, store incoming data at end of deque buffer
```

```
data = msg["data"]
for val in data :
    buf.append(val)
```

```
### Process
```

```
while True :
    if (len(buf) < 3) :
        break
    b = buf.popleft()
    if (isBeginOfFrame(b)) :
        b2=buf.popleft()
        b3=buf.popleft()
        index = b2 & 0x7f + ((b3 & 0x7f) << 7)
        height = index * 0.05 * 1000
```

```
out = {}
out["height"] = height
rep = [ out ]
```

**Advanced numerical computation : The kalman filter**

Use manta, python and graphs to simulate the response of your filters and design advanced low latency estimators like :

- Dynamic filter (True Wind damping)
- Sensors hybridation (Surface speed, Ride height)
- Estimate a constant, a position

**Simulate a noisy signal :**

The simulated measurement is generated with the uncertainty knowledge.

```

### Noisy sinus generator with outliers - may 9, 2019 - v1.0
###
### IN0 : Must be connected to a tick (Typ 25Hz)
### OUT0 : 'Signal' : Simulate a measurement (A noisy sinus with glitches)
###      'Covariance' : Covariance of the measurement
###
### Copyright © 2019 Pixel Sur Mer

import random

try :
    init
except NameError :
    init = True
    t=0.0
    cmpt=0
    out = {}

t=t+0.04
cmpt=cmpt+1
if cmpt>50 :
    cmpt=0
    offset=random.uniform(-50.0, 50.0)
else :
    offset=0.0
noise=random.gauss(0, 2)+offset      # gaussian noise + glitches
out["Signal"] = 10.0*math.sin(t)+noise

```

```
out["Covariance"] = noise**2          # In this case noise mean=0 -> covariance
rep = [ out ]
```

### **Dynamic Kalman filter**

The uncertainty of the measurement is used to reject noises. This is a generic implementation of a Kalman filter, you can use this template for another use of the kalman filter.

```
### Implementation of a dynamic Kalman Filter - may 9, 2019 - v1.0
###
### Matrices must be set up for your system
### Notation : https://en.wikipedia.org/wiki/Kalman\_filter
### Basic tutorial : https://www.kalmanfilter.net/default.aspx
###
### IN0 : 'Signal' :      Measurement (fix frequency)
###      'Covariance' : Optional covariance of this measurement
### OUT0 : 'KF_Predict' : Kalman predicted output
###
### Copyright © 2019 Pixel Sur Mer
```

```
import numpy as np
```

```
global np
```

```
class Kalman(object) :
```

```
    def __init__(self, F = None, B = None, H = None, Q = None, R = None, P = None) :
```

```
        if(F is None or H is None) :
```

```
            raise ValueError("Set proper system dynamics.")
```

```
        self.n = F.shape[1]
```

```
        self.m = H.shape[1]
```

```
        self.F = F
```

```
        self.H = H
```

```
        self.B = 0 if B is None else B
```

```
        self.Q = np.eye(self.n) if Q is None else Q
```

```
        self.R = np.eye(self.m) if R is None else R
```

```
        self.P = np.eye(self.n) if P is None else P
```

```
        self.x = np.zeros((self.n, 1)) if x0 is None else x0
```

```
    def predict(self, u = 0) :
```

```
        self.x = np.dot(self.F, self.x) + np.dot(self.B, u)
```

```
        self.P = np.dot(np.dot(self.F, self.P), self.F.T) + self.Q
```

```

        return self.x

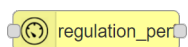
    def update(self, z, R = None) :
        self.R = np.eye(self.n) if R is None else R
        y = z - np.dot(self.H, self.x)
        S = self.R + np.dot(self.H, np.dot(self.P, self.H.T))
        K = np.dot(np.dot(self.P, self.H.T), np.linalg.inv(S))
        self.x = self.x + np.dot(K, y)
        I = np.eye(self.n)
        self.P = np.dot(np.dot(I - np.dot(K, self.H), self.P),
                        (I - np.dot(K, self.H)).T) + np.dot(np.dot(K, self.R), K.T)

    try :
        init
    except NameError :
        init = True
        out = {}
        dt = 1.0/60 # Time step
        cov = 0.5 # Default obser
        P = [] # Prediction
        F = np.array([[1, dt, 0], [0, 1, dt], [0, 0, 1]]) # State-transi
        H = np.array([1, 0, 0]).reshape(1, 3) # Observation
        Q = np.array([[0.05,0.05,0.0],[0.05,0.05,0.0],[0.0,0.0,0.0]]) # Covariance o
        R = np.array([cov]).reshape(1, 1) # Covariance o
        kf = Kalman(F = F, H = H, Q = Q, R = R)

    if "Covariance" in msg :
        cov=msg["Covariance"]
    if "Signal" in msg :
        pred=np.dot(H, kf.predict())[0] # Predicted st
        P.append(pred)
        kf.update(msg["Signal"], R=np.array([cov]).reshape(1, 1)) # Calc kalman
        out["KF_Predict"] = pred[0]
    else :
        raise ValueError("No data 'Signal'")

```

## 11.67 regulation\_performance

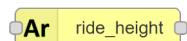




## Description

This box is a pilot overlay platform for implementation of multi-objective regulation/performance rule.

### 11.68 ride\_height



## Description

The purposes of this Manta Box is :

- To compute the height above the water (z axis) of points of interest, computed from some look down range sensor(s)
- The range sensor(s) shall measure a distance to the water which is pitch/roll independant => shortest distance (typical for sonic sensor)
- Up to 6 range measurement sensors can be configured. For outputs as many points can be configured where the height above water will be computed
- This Manta box is designed to work with SI unit only. (meters & deg)
- The heights outputs will have two values
  - One with extension : .csys => height in the specified coordinate system (see Properties)
  - One with extension : .disp => height always positive when point is above water plan (Z axis up)
- Compute height dynamic state to understand if height is increasing, decreasing or stable

## Properties

- **global properties** : Used for each sensors :
- **Name** : Manta box name
- **Filter duration** : Sensors filter duration in time unit
- **Min range** : Sensor measurement below this value will be rejected.
- **Max range** : Sensor measurement above this value will be rejected.
- **Coord. System** : Right Hand or SNAME (Right hand pointing down).
- **Slope Lag Duration** : Backward duration to define height dynamic state
- **Height Deadband** : Deadband in wich no need to compute height dynamic state
- **Sensor Count** : Number of sensor to compute

- **Sensor configuration :**

- Sensor variable name
- XYZ location in meter
  - \* RH : X+ => Fwd, Y+ => to Port, Z+ => Up
  - \* SNAME : X+ => Fwd, Y+ => to Stbd, Z+ => Down

- **Input mapping**

- Heel variable (RH & SNAME : positive => heel to stbd )
- Trim variable (RH : positive => bow down. SNAME : positive => bow up )

- **Output mapping**

- Height name : nmea of the output height
- Height Derivative Sign name :
- XYZ : Distance in meter of point of interest

## Inputs

- **Input 1 :**

- Heel variable (RH & SNAME : positive => heel to stbd )
- Trim variable (RH : positive => bow down. SNAME : positive => bow up )

- **Other input :**

- Sensors 1, 2, ...

## Outputs

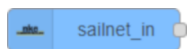
- **Output 1** As many as required output heights

- RH : X+ => Fwd, Y+ => to Port, Z+ => Up
- SNAME : X+ => Fwd, Y+ => to Stbd, Z+ => Down

- **Output 2** height dynamic state :

- Positive : Height increasing
- Negative : Height decreasing
- Zero : no change

## 11.69 sailnet



### Description

This Manta box imports NKE systems data by using SailNet protocol. To grab data from NKE *Processor HR* or *Processor Regatta*, the SaiNet protocol must be initiaized from NKE system web interface.

From « Outils » -> « Parcourir le répertoire courant du Processor », open « /mnt/flash/processor/instal/Instal.ini » :

- *ValidSailNet = Y*
- *SailNetOutIp = Exocet IP address*
- *SailNetOutPort = Manta box Local port*
- *SailNetInPort = Manta box Remote port*
- *Save File when done*

From « Développement à façon » -> « Paramétrage de SailNet » :

- *SailNetOutIp = Exocet IP address*
- *SailNetOutPort = Manta box Local port*
- *SailNetInPort = Manta box Remote port*
- *Save File when done*

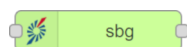
### Properties

- **Name** : Manta box name
- **NKE system IP address** : Enter IP address of the NKE device.
- **Local port** : UDP port to receive data. It corresponds to *SailNetOutPort*.
- **Remote port** : NKE UDP port to send query. It corresponds to *SailNetInPort*.
- **NKE variables to export** : Select from the list the NKE variables to import. Note that the Manta graph must have been deployed a first time after adding the *NKE* box to populate the variables list.

### Outputs

- The list of selected NKE variables (Integer or Decimal)

## 11.70 sbg



### Description

This Manta box decode some SBG Inertial Navigation System binary messages.

Supported messages are : LOG\_STATUS (01), LOG\_UTC\_TIME (02), LOG\_IMU\_DATA (03), LOG\_EKF\_EULER (06), LOG\_EKF\_NAV (08), LOG\_SHIP\_MOTION (09), LOG\_GPS1\_VEL (13), LOG\_GPS1\_POS (14), LOG\_GPS1\_HDT (15)

**Vessel reference frame :** The data provided by the Sbg Manta box use the standard SNAME's (1950) notation (forward/starboard/downward and North/East/Down frames).

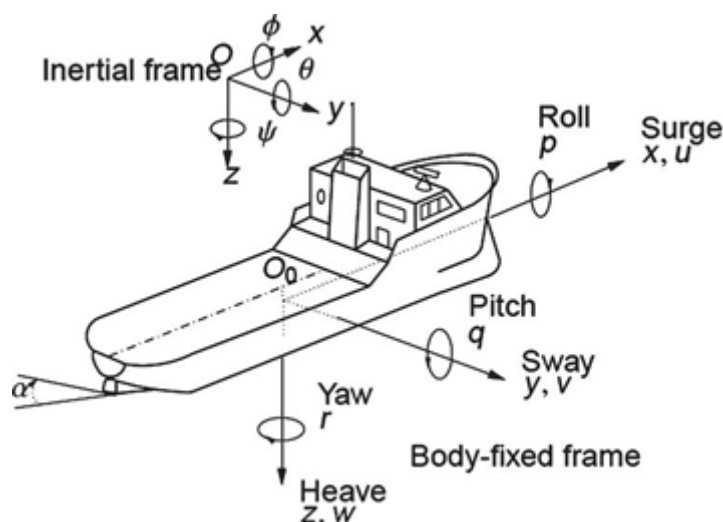


FIG. 86

### Properties

- **Name** : Manta box name
- **Msg ID xx** : If selected, decode this message

### Inputs

- **Input** : UDP or serial stream to decode.

## Outputs

- **01-LOG\_STATUS** : Output data of SBG\_ECOM\_LOG\_STATUS (01)
- **02-LOG\_UTC\_TIME** : Output data of SBG\_ECOM\_LOG\_UTC\_TIME (02)
- **03-LOG\_IMU\_DATA** : Output data of SBG\_ECOM\_LOG\_IMU\_DATA (03)
- **06-LOG\_EKF\_EULER** : Output data of SBG\_ECOM\_LOG\_EKF\_EULER (06)
- **08-LOG\_EKF\_NAV** : Output data of SBG\_ECOM\_LOG\_EKF\_NAV (08)
- **09-LOG\_SHIP\_MOTION** : Output data of SBG\_ECOM\_LOG\_SHIP\_MOTION (09)
- **13-LOG\_GPS1\_VEL** : Output data of SBG\_ECOM\_LOG\_GPS1\_VEL (13)
- **14-LOG\_GPS1\_POS** : Output data of SBG\_ECOM\_LOG\_GPS1\_POS (14)
- **15-LOG\_GPS1\_HDT** : Output data of SBG\_ECOM\_LOG\_GPS1\_HDT (15)

## 11.71 scalar



### Description

User controlled box.

This Manta box exports a scalar data that can be changed from Exocet Web App (*Integer or Decimal*).

### Properties

- **Name** : Manta box name
- **Save** : If selected, the data value is saved to be restored when Exocet is rebooted or Manta graph deployed.
- **Output Name** : Exported data name. Manta box name is used if not defined.
- **Period** : to send data value periodically. Data value is also sent at any user change. If *None* is selected, only data changes are sent.
- **Type** :
  - **Integer Range** : an Integer value within the range bordered by *min* and *max*. Any value within the range is possible.
  - **List of defined Integer values** : an integer value that can takes only a set of pre defined values.
  - **Decimal Range** : a Decimal value within the range bordered by *min* and *max*. Any value within the range is possible.

- **List of defined Decimal values** : a decimal value that can takes only a set of pre defined values.
- **Decimals** : number of decimals to display the value.
- **Initial Value** : initial data value.
- **Step Value** : step applied when incrementing or decrementing data from web application.
- **Min Value** : minimum value for the data
- **Max Value** : maximum value for the data
- **List of defined data values** : Press *New predefined value* button to fill a new value. For each value, a *Label* can be defined to help web application support. The initial value is the first defined value.
- **Ext control** : If selected, add inputs to externally control the scalar value.
- **Unit** : select unit of the exported data

### Inputs

- No Input connector, the input comes from Web App.
- **Scalar** : If *Ext control* is selected, a scalar value can be received to set the data value.
- **Minus** : If *Ext control* is selected, a boolean rising edge can be received to decrement the data value.
- **Plus** : If *Ext control* is selected, a boolean rising edge can be received to increment the data value.
- **Reset** : If *Ext control* is selected, a boolean rising edge can be received to return to the *Initial Value*.

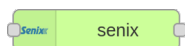
### Outputs

- **Output Name** : Integer or decimal data value

### WebApp controls

- **Output Name** (*int or float*) : On your dashboard, use a « Set number » or « Slider » widget to change value.

### 11.72 senix



## Description

Decode Senix ASCII protocol data sent via serial interface.

## Properties

- **Name** : Manta box name
- **Sensor type** : Select product type
- **Output name** : Exported altitude data name
- **Add status** : If selected, check min/max optimum & extremum validity. Export result as a bitfield (see *Notes* below) and as a status icon.

## Inputs

- **serialin** : serial data stream

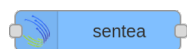
## Outputs

- **data** : altitude data + optional data status (see note below)

## Notes

Status convention : \* bit 0 : ON/OFF (0=off 1=on) \* bit 1 : Problem detected (0=valid 1=error) \* bit 2 : Problem severity (0=warning 1=error)

## 11.73 sentea



## Description

Export variables such as microstrains or temperatures from Sentea optical interrogator using UDP mode, format 202. It is compatible with DM-4120 and DM-8120 interrogators.

The Sentea box is configured using files generated by Sentea *Peakviewer* software. This software must be used to configure the optical interrogator (see *Notes* below).

## Properties

- **Name** : Manta box name
- **Local port** : Enter the UDP input local port
- **Remote IP** : Enter interrogator IP address
- **CSV file** : Select one of the already downloaded .csv file, containing sensor definitions
- **INI file** : Select one of the already downloaded .ini file, containing sensor equations
- **Export peaks** : Check to export peaks on Debug output
- **Upload new Conf** : Upload a new *moi* file from your PC to the Exocet
- **Delete or Download configuration files** : Use *Trash icon* to delete a configuration file and the *download icon* to download the configuration file from the Exocet to the PC
- **Select variables** : Check/Uncheck the variables to export out of the Manta box

## Inputs

- **Enable** (*bool*) : Connect a boolean to enable/disable the box. Default is enable.

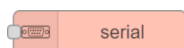
## Outputs

- **Data** : A set of variables such as microstrains or temperatures according to the configuration file.
- **Debug** : Some useful data.

## Notes

The Sentea *Peakviewer* software must be used to configure the optical interrogator and to define sensors : \* Configure IP address to be in the same subnet. \* Write and load a *python script* to configure UDP output communication (format 202), capture and peakfind parameters. Use the *auto start* option to automatically execute the script at startup of the interrogator. \* Use the *Sensor definition editor* to configure each FBG parameters (name, wavelength min/ref/max). Export these definitions to a csv file, and upload it directly on the Exocet. \* Use the *Sensor equation editor* to define each constants and sensor equations. FBG and reference names (*\_0*) come from *sensor definitions*. All constants and variables are shared and can be used in any equation. The *Unit* field should be one the following : *Strain, Temperature, Pressure, Acceleration, Displacement, Wavelength, Angle*. Export these equations to the *ini* file, and upload it directly on the Exocet.

## 11.74 serial





## Description

Send data to serial port. Exocet sends data to the specified RS232 or RS422 port.

## Properties

- **Name** : Manta box name
- **Serial port** : Serial port configuration box. Press the pencil icon to create a new one. Fill the serial line parameters, then press the *Add* button. Once created, it can be shared with other Serial in or out boxes.
- **WebApp** : Allow port selection from Exocet WebApp dashboards

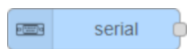
## Inputs

- Data to send. Must be of *Opaque* or *String* type. Note that string variables are limited to 32 characters.

## Notes

COM3 to COM6 use an USB extension, behaviour can differ from COM1 and COM2.

## 11.75 serial\_in



## Description

Get data from a serial port. Exocet received data from the specified RS232 or RS422 port.

## Properties

- **Name** : Manta box name
- **Serial port** : Serial port configuration box. Press the pencil icon to create a new one. Fill the serial port parameters, then press the *Add* button. Once created, it can be shared with other Serial in or out boxes.
- **Add TS** : Add timestamp to output data flow
- **WebApp** : Allow port selection from Exocet WebApp dashboards

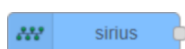
## Outputs

- **data** (*Opaque*) : received serial data. Depending of use, a buffering may be necessary.

## Notes

COM3 to COM6 use an USB extension, behaviour can differ from COM1 and COM2.

### 11.76 sirius



## Description

Get Multiple Sensors from Pixel Sur Mer Sirius gateway.

Sirius is a GPIO sensors gateway used for different purpose : water intrusion detection, sail and rudder position detection, hook position ... It is based on 1 wire technology. A Sirius Gateway can handle up to 3 sensor buses. Each bus can handle multiple sensors.

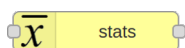
## Properties

- **Name** : Manta box name
- **Local port** : Port to received data from

## Outputs

- A list of sensors *Boolean* values.

### 11.77 stats



## Description

This Manta Box can perform average, standard deviation, min/max search and others on a data dictionary.

## Properties

- **Name** : Manta box name
- **List of processes** :
  - **Process type** : Select the desired process type (see *Notes* below).
  - **Output name** : Exported data name (optional).
  - **Percent** : Only for percentile process type, set the threshold for the percentile process.

## Inputs

- **Data** : Connect here the data dictionary to process. Only one wire should be connected on Data input.

## Outputs

- **Processed\_data** : List of results. *Float* type, except for names.

## Notes

### Process type :

- **Average** : Get the sum of data divided by the number of data.
- **Variance** : Return how far the set of data are spread out from their mean value.
- **Standard deviation** : Return the dispersion of data from the mean value.
- **Mean deviation** : Return the average dispersion of data from the mean value.
- **Min** : Get the min value from the current set of data, and its name.
- **Max** : Get the max value from the current set of data, and its name.
- **Median** : Get the middle value from the current set of data, i.e. the value that divide the set in two equal parts.
- **Percentile** : Get the given percentile value from the current set of data, i.e. the value below which the given percentage falls (0% gets the minimum, 100% gets the maximum, 50% gets the median).
- **Range** : Get the *Max* minus *Min* value.
- **Max of abs** : Get the max of absolute values of data, and its name.
- **Sum** : Get the sum of data

## 11.78 string



### Description

User controlled box.

This Manta box exports a string data that can be changed from Exocet Web App.

### Properties

- **Name** : Manta box name
- **Save** : If selected, the string data is saved to be restored when Exocet is rebooted or Manta graph deployed.
- **Ext control** : If selected, add inputs to externally control the string value.
- **Output Name** : Exported data name. Manta box name is used if not defined.
- **Period** : to send data string periodically. String is also sent at any user change. If *None* is selected, only value changes are sent.
- **Type** : select *single* or *list of string*
- **Initial Value** : (Single type only) initial data value
- **List of strings** : (List of string type only) Press *New predefined value* button to fill a new value. The initial value is the first defined value. Maximum is 50.

### Inputs

- No Input connector, the input comes from web app Dashbord widget
- **String** : If *Ext control* is selected, a string value can be received to set the data value.
- **Reset** : If *Ext control* is selected, a boolean rising edge can be received to return to the *Initial Value* or the initial *List of strings*.
- **Add list option** : If *Ext control* and *list of string* is selected, a string value can be received to add a choice in the list of strings
- **Remove list option** : If *Ext control* and *list of string* is selected, a string value can be received to remove a choive in the list of strings

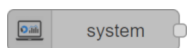
### Outputs

- **Output Name** : String data value

## WebApp controls

- **Output Name** (*string*) : On your dashboard, use a « Set Text » widget to change value.

## 11.79 system



### Description

This Manta box exports system usage data. Some of these data are only exported when used.

### Properties

There is no customizable properties.

### Outputs

Data Name	Unit	Description
<b>Bus Load Can X</b>	%	Estimated Can X bus load
<b>COMX Rx Load</b>	%	Estimated COM X load of the reception
<b>COMX Tx Load</b>	%	Estimated COM X load of the transmission
<b>Conf usage</b>	%	Estimated configuration memory usage (Manta graph, dashboards, user files...)
<b>CPU X</b>	%	Estimated load of CPU X
<b>CPU Total</b>	%	Estimated load of all CPU
<b>CPU Temperature</b>	°C	System CPU temperature
<b>Cycle usage</b>	%	Max time load of main periodic thread
<b>Ethernet</b>	kB/s	Ethernet data transfert rate
<b>Manta loop detected</b>	-	Data loop detected in the Manta graph
<b>Mem Available</b>	Mb	Estimated available RAM memory
<b>Mem Total</b>	Mb	Total amount of usable RAM memory

Data Name	Unit	Description
<b>Mem usage</b>	%	Ratio of available memory on total memory
<b>Node input overflow</b>	-	Total number of Node input buffer overflow from start-up
<b>Num CPU</b>	-	CPU number
<b>Num variables usage</b>	%	Overall number of variables on maximum number
<b>Output Connectors usage</b>	%	Overall number of box outputs on maximum number
<b>Record disk usage</b>	%	Estimated record memory usage
<b>Record files usage</b>	%	Number of record files on maximum number
<b>Slowdown detected</b>	-	System slowdown detected
<b>Status_CanX</b>	-	CAN bus overload detected
<b>Status_ComXRx</b>	-	Reception COM X overload detected
<b>Status_ComXTx</b>	-	Transmission COM X overload detected
<b>Status_ConfUsage</b>	-	Configuration memory usage too high detected
<b>Status_CpuX</b>	-	CPU X overload detected
<b>Status_CpuTotal</b>	-	CPU total overload detected
<b>Status_CpuTemperature</b>	-	CPU temperature too high detected
<b>Status_CycleUsage</b>	-	Cycle usage too high detected
<b>Status_MemUsage</b>	-	RAM memory usage too high detected
<b>Status_NumVariablesUsage</b>	-	Too many variables detected
<b>Status_OutConnectorsUsage</b>	-	Too many output connectors detected
<b>Status_RecordsDiskUsage</b>	-	Record memory usage too high detected
<b>Status_RecordsFilesUsage</b>	-	Record files usage too high detected
<b>Status_WebRefresh</b>	-	Too long web refresh period detected
<b>System status</b>	-	Overall system status (see note below)
<b>System uptime</b>	s	Exocet application uptime
<b>Voltage too low</b>	-	Total number of low power detected

Data Name	Unit	Description
<b>Voltage too low detected</b>	-	Low power detected
<b>Web refresh</b>	ms	Max refresh delay of WebApp

Output frequency : 0.5 Hz

## Notes

**CPU** Load is the ratio of active time on active plus idle time. It is calculated on 2s. Total is the average of all cores.

**Variables** Max allowed variables is 20,000. Max allowed output connectors is 2048.

**Status** Status convention : \* bit 0 : ON/OFF (0=off 1=on) \* bit 1 : Problem detected (0=valid 1=error) \* bit 2 : Problem severity (0=warning 1=error)

## 11.80 table



## Description

User controlled box.

This Manta box exports a table data that can be changed from Exocet Web Application. Maximum size is 32 x 32.

## Properties

- **Name** : Manta box name
- **Save** : If selected, the data value is saved to be restored when Exocet is rebooted or Manta graph deployed.
- **Output Name** : Exported data name. Manta box name is used if not defined.
- **Period** : to send data value periodically. Data value is also sent at any user change. If *None* is selected, only data changes are sent.

- **Ext control** : If selected, add inputs to externally control the table value.
- **Use text file** : If selected, use selected *Table file* as initial table. Then, all changes in table are saved in the file.
- **Initial table** : Enter initial values of the table (integer or float format). Column separator must be space or tab.
- **Table file** : Select one of the already downloaded Table files. It must be text file with extension .txt. Column separator must be space or tab.

## Inputs

- No Input connector, the input comes from Web App.
- **Table** : If *Ext control* is selected, a table can be received to set the table.
- **Reset** : If *Ext control* is selected, a boolean rising edge can be received to return to the *Initial table*.

## Outputs

- **Output Name** : Table data value

## WebApp controls

- **Output Name** (*float table*) : On your dashboard, use a « Table input » widget to change values.

## 11.81 tcp\_in



## Description

Get data from a TCP interface.

## Properties

- **Name** : Manta box name
- **Client** : Check to work as a client, otherwise it works as a server.
- **Local port** : Enter the input local port

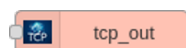


- **Remote IP** : Enter the emitter IP address
- **Remote port** : Enter the emitter output port
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards
- **Add TS** : Add timestamp to output data flow

## Outputs

- **data** (*Opaque*) : TCP received data

### 11.82 tcp\_out



## Description

Send data to a TCP interface.

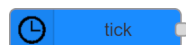
## Properties

- **Name** : Manta box name
- **Client** : Check to work as a client, otherwise it works as a server.
- **Local port** : Enter the output local port
- **Remote IP** : Enter the receiver IP address
- **Remote port** : Enter the receiver input port
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards

## Inputs

- Data to send. Must be of *Opaque* or *String* type. Note that string variables are limited to 32 characters.

### 11.83 tick



## Description

This box generates a periodic tick. It can for example be used to periodically run a Python box.

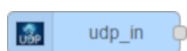
## Properties

- **Name** : Manta box name
- **Frequency** : Select the tick event frequency. **IMPORTANT NOTE** : Selecting a frequency up to 50Hz may impact Exocet CPU performances.

## Outputs

- **count** (*Integer*) : Tick counter
- **period** (*Float*) : Period measurement in ms
- **frequency** (*Float*) : Inverse of period measurement in Hz

## 11.84 udp\_in



## Description

Get data from an UDP interface.

## Properties

- **Name** : Manta box name
- **Local port** : Enter the input local port
- **Add TS** : Add timestamp to output data flow
- **Filter** : If selected, filter incoming data on local port using *Filter IP*
- **Filter IP** : Only if *Filter* is selected, IP to filter out
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards

## Outputs

- **data** (*Opaque*) : UDP received data

### 11.85 udp\_out



#### Description

Send data to an UDP interface.

#### Properties

- **Name** : Manta box name
- **Local port** : Enter the output local port (optional)
- **Remote IP** : Enter the receiver IP address
- **Remote port** : Enter the receiver input port
- **Broadcast** : Allow data to be sent to a broadcast address
- **WebApp** : Allow connexion parameters access from Exocet WebApp dashboards

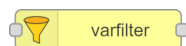
#### Inputs

- Data to send. Must be of *Opaque* or *String* type. Note that string variables are limited to 32 characters.

#### WebApp controls

- **Remotelp** (*string*) : If *WebApp* is selected, on your dashboard, use a « Set Text » widget to change value.
- **RemotePort** (*int*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change value.

### 11.86 varfilter



#### Description

The purposes of this Manta Box is to extract one or more variables from an input data stream.

## Properties

- **Name** : Manta box name
- **Splitter** : Add an output to export rejected variables
- **Variables** : Select variables to extract. Note that the Manta graph must have been deployed a first time after adding the *VarFilter* box and connecting it, to populate the variables list.

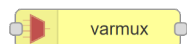
## Inputs

- **Data stream** : Data stream to analyse

## Outputs

- **Filter\_in** : Extracted variables
- **Filter\_out** : Rejected variables, only if *Splitter* option has been checked

## 11.87 varmux



## Description

The purpose of this Manta Box is to multiplex several input data streams.

## Properties

- **Name** : Manta box name
- **Frequency** : Select output data frequency :
  - Value of the desired output data frequency.
  - *slowest* to synchronise with the slowest input data stream received on *Main input* (see *Notes* below).
  - *fastest* to send each time an input data is received on *Main input*.
- **Policy** : Select the desired sending policy (only applied to data streams from the *Main input*) :
  - *Standard* : Send data according to the selected frequency.
  - *Send only if refreshed* : Send data according to the selected frequency if at least one variable from the *Main input* has been updated.

- *Send immediately if refreshed* : Send data according to the selected frequency and when at least one variable from the *Main input* has been updated.
- *Send only if changed* : Send data according to the selected frequency if at least one variable from the *Main input* has changed.
- *Send immediately if changed* : Send data according to the selected frequency and when at least one variable from the *Main input* has changed.
- **Pass through** : Export all non renamed variables, permit to filter out all non renamed variables
- **Disable checks** : Disable configuration check at runtime to save CPU
- **List of variables to rename** (optional) :
  - **Var\_in** : Enter the input variable name
  - **Var\_out** : Enter the name to use at output

## Inputs

- **Main input** : Data streams to be multiplexed (max 64) and be used by the selected *policy* and *frequency*
- **Secondary input** : Other data streams to be multiplexed (max 64)

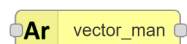
## Outputs

- Data stream containing all data of input data streams, and sent at the specified *frequency* and *policy*.

## Notes

- Maximum number of variables is limited to 1024.
- The principle of *slowest* frequency is to update the output when all data streams connected to the *Main input* have been refreshed. This can block the output if one data stream is no longer refreshed. To avoid this, a monitoring algorithm update the output if the delay is too long compare to the last update periods. To disable this algorithm you can select the *Send only if refreshed* policy.
- You can also use the *Main input* timeout to stop sending if main input is not refreshed.

## 11.88 vector\_man



## Description

The purposes of this Manta Box is :

- To rotate a 3D vector by providing 3 rotations angles for each axis
- You can scale each axis component with individual factors
- You can add an offset to each axis component

## Properties

- **Name** : Manta box name
- **Pass through** : output vector = input vector
- **Frequency** : Box processing frequency
- **Rotation order** : RPY => roll pitch yaw, YRP => yaw pitch roll
- **Mult. factors** : x y z multiplying factors
- **Offsets** : x y z offsets
- **Inputs mapping**
  - **X** : X vector name.
  - **Y** : Y vector name.
  - **Z** : Z vector name.
  - **Heel** : Trim name.
  - **Trim** : Trim name.
  - **Heading** : Heading name.
- **Outputs mapping**
  - **New vector** : A generic name for the new vector.

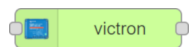
## Inputs

- Input 1 : Vector to be manipulated (XYZ).
- Input 2 : Heel, Trim and Heading. If no data is provided no rotation will be applied to that axis

## Outputs

- 3 outputs data will be generated with extension .x .y .z

## 11.89 victron



### Description

Decode Victron VE.Direct-Protocol data sent via serial interface.

### Properties

- **Name** : Manta box name

### Inputs

- **serialin** : serial input frame

### Outputs

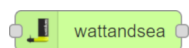
- **data** : device data

Data are exported each second.

Data Name	Description	Type	Unit
<b>vbat</b>	Main battery voltage	Float	Volts
<b>vaux</b>	Auxiliary (battery) voltage	Float	Volts
<b>vpan</b>	Panel voltage	Float	Volts
<b>wpan</b>	Panel power	Integer	Watt
<b>ibat</b>	Battery current	Float	Amperes
<b>iload</b>	Load current	Float	Amperes
<b>temp</b>	Battery temperature	Integer	°C
<b>pow</b>	Instantaneous power	Integer	Watt
<b>consumedAmpHours</b>	Consumed amp hours	Float	Amperes Hour
<b>soc</b>	State of charge	Float	%
<b>timeToGo</b>	Time to go	Integer	Minutes

Data Name	Description	Type	Unit
<b>alarm</b>	Alarm condition active	Boolean	N/A
<b>relay</b>	Relay state	Boolean	N/A
<b>alarmreason</b>	Alarm reason	Integer	N/A
<b>deepestDischarge</b>	Depth of the deepest discharge	Float	Amperes Hour
<b>nbChargeCycles</b>	Number of charge cycles	Integer	N/A
<b>vbatMin</b>	Minimum main (battery) voltage	Float	Volts
<b>vbatMax</b>	Maximum main (battery) voltage	Float	Volts
<b>vauxMin</b>	Minimum auxiliary (battery) voltage	Float	Volts
<b>vauxMax</b>	Maximum auxiliary (battery) voltage	Float	Volts
<b>maxpowtoday</b>	Maximum power today	Integer	Watt
<b>errorCode</b>	Error code	Integer	N/A
<b>stateOfOperation</b>	State of operation	Integer	N/A

## 11.90 wattandsea



### Description

Decode Watt&Sea status sent via serial interface

### Properties

- **Name** : Manta box name
- **Protocol** : Select the desired protocol

### Inputs

- **serial\_in** : serial frame to decode



## Outputs

### Aero Generator :

- **Data :**

Data Name	Description	Type	Unit
<b>RPM</b>	Generator speed	Integer	RPM
<b>VInDC</b>	Input voltage	Float	Volts
<b>VCharge</b>	Battery charge Voltage	Float	Volts
<b>IchBat1</b>	Battery 1 current	Float	Amperes
<b>IchBat2</b>	Battery 2 current	Float	Amperes

- **System :**

Data Name	Type	Unit
<b>VBus</b>	Float	Volts
<b>IBus</b>	Float	Amperes
<b>PBus</b>	Float	Watts
<b>VchBat1</b>	Float	Volts
<b>VchBat2</b>	Float	Volts
<b>Temperature</b>	Float	°C

### Aero-v2 Generator :

- **Converter :**

Data Name	Description	Type	Unit
<b>Vgene</b>	Generator voltage	Float	Volts
<b>Vbatt</b>	Battery voltage	Float	Volts
<b>Ibatt</b>	Battery current	Float	Amperes
<b>RPM</b>	Generator speed	Integer	RPM

Data Name	Description	Type	Unit
<b>DefBatt</b>	Battery default	Integer	-
<b>EtatCharge</b>	Charge state	Integer	-
<b>Alarme</b>	Alarms	Integer	-

- **Servo :**

Data Name	Type	Unit
<b>Current</b>	Float	Amperes
<b>Velocity</b>	Integer	-
<b>Position</b>	Float	Deg
<b>Voltage</b>	Float	Volts
<b>Temperature</b>	Integer	°C

### Hydro Generator :

- **Converter :**

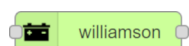
Data Name	Description	Type	Unit
<b>Uin</b>	Generator voltage	Float	Volts
<b>Uout</b>	Output voltage	Float	Volts
<b>Iout</b>	Output current	Float	Amperes
<b>Temp</b>	Converter internal temperature	Float	°C
<b>F1</b>	Generator #1 speed	Integer	RPM
<b>F2</b>	Generator #2 speed	Integer	RPM

- **Pump :**

Data Name	Type	Unit
<b>Pressure</b>	Float	Bars

Data Name	Type	Unit
<b>ModeAuto</b>	Bool	-
<b>Pitch</b>	String	-
<b>Active</b>	Bool	-

## 11.91 williamson



### Description

Decode williamson battery status sent to CAN bus

### Properties

- **Name** : Manta box name
- **Reference** : select battery reference (see note below)
- **Refresh** : data refresh period in second. From 1 to 3600.
- **Battery ID** : select the ID of the battery. From 1 to 25.
- **Use BatID as prefix** : If selected, all battery data are prefixed with « BatX\_ ». \* \*\*

### Inputs

- **canin** : can bus input to receive battery data

### Outputs

- **data** : battery data
- **canout** : can bus output to send battery status query

Each battery sends the following data :

Data Name	Description
<b>fault</b>	Fault code. See below

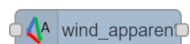
Data Name	Description
<b>ich</b>	Charge current
<b>iequil</b>	Balancing current
<b>iload</b>	Load current
<b>levelch</b>	Charge level
<b>nbcycles</b>	Cycle numbers
<b>tbat</b>	Internal temperature
<b>tmos</b>	Switches temperature
<b>vbat</b>	Battery voltage
<b>vch</b>	Charge voltage

**Fault codes :**

- **0** : Balancing current hight
- **2** : Discharge current hight
- **4** : User overvoltage
- **20** : Charging current hight
- **64** :User short-circuit
- **68** : overvoltage or temperature hight

**Notes****Battery Reference**

- **WILPA1741, WILPA2044, WILPA2533A** : with SAFT element of type MP176065 G5 and G6
- **WILPA2795A** : with SAFT element of type MP176065 xlr
- **WILPA2935A** : with Panasonic element of type NCR 18650A

**11.92 wind\_apparent**

## Description



**FIG. 87**

Back-calculation of the undamped apparent wind in the horizontal plane using True wind and navigation data. Only one calculation is available. All inputs are necessary.

## Properties

- **Name** : Manta box name
- **Ref. to Course** : The Apparent Wind Angle is referenced to the Course axis. Unchecked : Referenced to the longitudinal boat axis
- **TW ref. to Course** : The True Wind Angle is referenced to the Course axis. Unchecked : Referenced to the longitudinal boat axis

## Inputs

### Input 0

Name	Units	Description
<b>TWA_undamp</b>	(°T)	Undamped true wind angle, referenced on the horizontal plane
<b>TWS_undamp</b>	(kn)	Undamped apparent wind speed, referenced on the horizontal plane

### Input 1

Name	Units	Description
<b>Leeway</b>	(°)	Undamped Leeway Angle

### Input 2

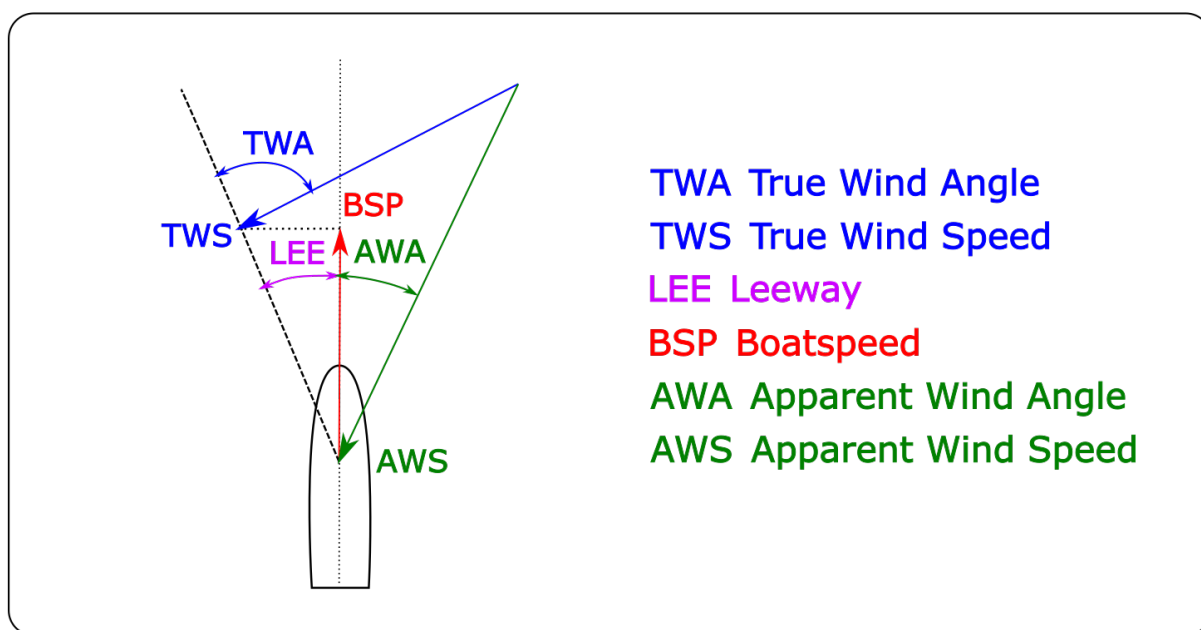
Name	Units	Description
<b>BoatSpeed</b>	(kn)	Undamped BoatSpeed relative to the surface of water

## Outputs

Name	Units	Description
<b>AWA_undamp</b>	(°)	Undamped Apparant wind angle, referenced on the horizontal plane
<b>AWS_undamp</b>	(°)	Undamped Apparant wind speed, referenced on the horizontal plane

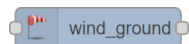
## Notes

### Apparent Wind definition :



**FIG. 88**

### 11.93 wind\_ground



#### Description



FIG. 89

Calculation of the ground wind using True Wind data and current, relative to the sea floor (referenced at 10m altitude if weather forecast comparison is required). Only one calculation is available. All inputs are necessary. No options are required.

#### Properties

- **Name** : Manta box name

#### Inputs

##### Input 0

Name	Units	Description
<b>TWD_disp</b>	(°T)	Damped True Wind Direction plane, relative to the sea surface and referenced on the horizontal plane
<b>TWS_disp</b>	(kn)	Damped True Wind Speed, relative to the sea surface and referenced on the horizontal plane

##### Input 1

Name	Units	Description
<b>TideSet</b>	(°T)	Damped Surface Current Direction

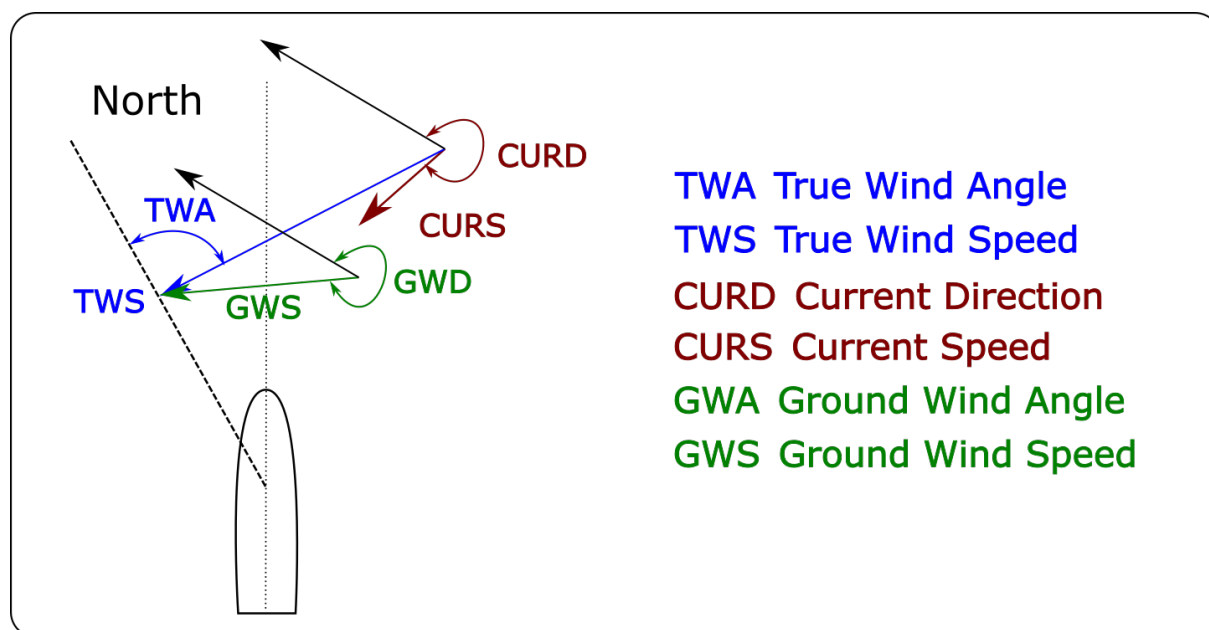
Name	Units	Description
<b>TideRate</b>	(kn)	Damped Surface Current Speed

## Outputs

Name	Units	Description
<b>GW_direction</b>	(°T)	Damped Ground Wind Direction, relative to the sea floor and referenced on the horizontal plane
<b>GW_speed</b>	(kn)	Damped Ground Wind Speed, relative to the sea floor and referenced on the horizontal plane

## Notes

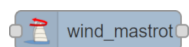
### Ground Wind definition :



**FIG. 90**



## 11.94 wind\_mastrot



### Description



**FIG. 91**

This Manta box compensate the selected Mast Head Unit « MHU\_angle » from the mast rotation and twist.

### Properties

- **Name** : Manta box name
- **Bypass** : Checked = No rotating mast in the installation
- **UseTwist** : Checked = Use twist correction

### Inputs

#### Input 0 : Calibrated Selected Mast Head Unit

Value Name	Units	Description	Calculations using this input
<b>MHU_angle</b>	(°)	Measured Wind Angle in 2D plane at the top of the mast	Measured Wind Angle (Output 0)
<b>MHU_speed</b>	(°)	Measured Wind Speed in 2D plane at the top of the mast	Measured Wind Speed (Output 0)

#### Input 1 : Optional Mast Rotation

Value Name	Units	Description	Calculations using this input
<b>RotatingMastAngle</b>	(°)	Mast Rotation Angle around the mast longitudinal axis and measured at the bottom of the mast, positive in clockwise direction	Measured Wind Angle (Output 0)

Note : Add a time out in this port if you use a mast angle sensor

#### Input 2 : Optional Mast Twist

Value Name	Units	Description	Calculations using this input
<b>MastTwist</b>	(°)	Mast Twist Angle measured between the bottom and the top of the mast, positive in clockwise direction	Measured Wind Angle (Output 0)

Note : Add a time out in this port if you use a mast twist solution

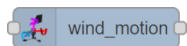
## Outputs

#### Output : Measured Wind

Value Name	Units	Description	Necessary inputs to calculate this data
<b>MW_angle</b>	(°)	Measured Wind Angle in 2D plane at the top of the mast	- MHU_angle (Input 0) - RotatingMastAngle (Input 1) - MastTwist (Input 2)
<b>MW_speed</b>	(°)	Measured Wind Speed in 2D plane at the top of the mast	- MHU_speed (Input 0)

Note : Output frequency is the same as Input 0 frequency

## 11.95 wind\_motion



### Description



FIG. 92

Calculate the Uncorrected Apparent Wind defined in the horizontal plane and corrected from :

- Hull and mast attitude effect (geometric tilt error on the 2D measured wind angle, and quasi cosine tilt response of the 2D anemometer sensor).
- The motion of the Mast Head Unit (due to the boat motion in the waves and to the big lever arm of the mast).

The Apparent Wind can be then back calculated from the upwash and wind shear corrected true wind.

### Properties

- **Name** : Manta box name
- **Correction** : define the level of wind denoising :
  - **Attitude** : Only hull and mast attitude (heel, trim, MastRake, MastCant) are applied to the measured wind projection on the horizontal plane. No dynamic motion compensation.
  - **Dual rotation** : Use pitch & roll angular velocities (p, q) and Mast Head Unit height (-MHUz) for motion correction.
  - **Attitude & dual rotations** : Use hull and mast attitude (heel, trim, MastRake, MastCant) for geometric correction, pitch & roll angular velocities (p, q) and Mast Head Unit height (-MHUz) for motion correction.
  - **Attitude & triple rotations** : Use hull and mast attitude (heel, trim, MastRake, MastCant) for geometric correction, pitch & roll & yaw angular velocities (p, q, r) and Mast Head Unit

full position (MHUx, MHUy, -MHUz) for motion correction. Useful in fast tacking or if the Mast Head Unit is not installed above the center of buoyancy.

- **Full Motion** : Use hull and mast attitude (heel, trim, MastRake, MastCant) for geometric correction, pitch/roll/yaw angular velocities (p, q, r) and surge/sway linear velocities (u, v) and Mast Head Unit full position (MHUx, MHUy, -MHUz) for motion correction. Useful if the position of the center of buoyancy can change dynamically (multihull).
- **MHU x,y,z** : Position of the mast head unit from the center of rotation (note : MHUz = -MHU altitude)
- **WebApp** : Allow MHU position access from Exocet WebApp dashboards

## Inputs

### Input 0

Value Name	Units	Description	Calculations using this input
<b>MW_angle</b>	(°)	Measured Wind Angle in 2D plane at the top of the mast	CMW_angle, CMW_speed (output 0)
<b>MW_speed</b>	(kn)	Measured Wind Speed in 2D plane at the top of the mast	CMW_angle, CMW_speed (output 0)

### Inputs 1

Name	Units	Description
<b>Heel</b>	(° or rad)	Boat attitude relative to the horizontal plane
<b>Trim</b>	(° or rad)	Boat attitude relative to the horizontal plane
<b>p, q, r</b>	(°/s or rad/s)	Angular velocities around x,y,z boat axis
<b>u, v, w</b>	(m/s)	Linear velocities in x,y,z boat axis

### Inputs 2

Name	Units	Description
<b>MastRake</b>	(°)	Mast attitude relative to the hull axis
<b>MastCant</b>	(°)	Mast attitude relative to the lateral axis

## Output

Name	Units	Description
<b>CMW_angle</b>	(°)	Corrected Measured wind angle (referenced on the horizontal plane passing through the MHU position)
<b>CMW_speed</b>	(°)	Corrected Measured wind speed (referenced on the horizontal plane passing through the MHU position)
<b>CMW_altitude</b>	(m)	Altitude of the wind measurement
<b>CMW_altRatio</b>	(%)	Equal to MHU_Height / Altitude (depending of the hull & mast altitude)

## WebApp controls

- **MHU x,y,z** (*float*) : If *WebApp* is selected, on your dashboard, use a « Set Number » widget to change MHU position.

## Note

**Reference :** The standard SNAME's (1950) notation (forward/starboard/downward and North,East,Down frames) is used internally and for sensor position. The origin of the position must be close to the center of rotation (top of the keel).

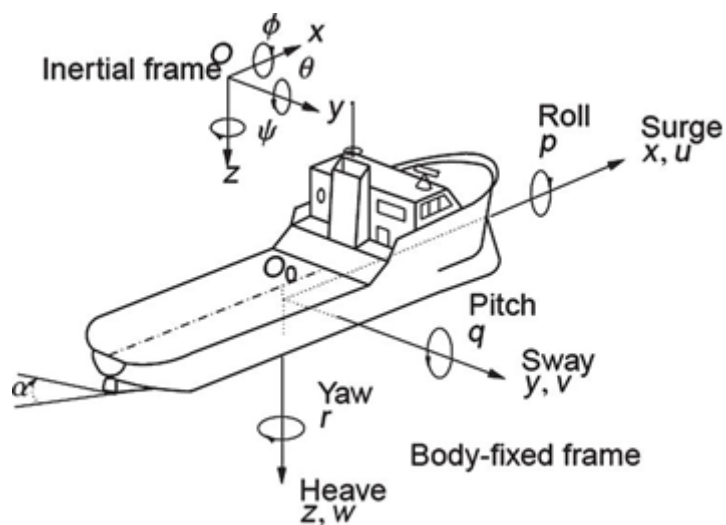
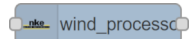


FIG. 93

## 11.96 wind\_processor\_hr



### Description

Encode data to drive Processor HR with Pixel wind.

### Inputs

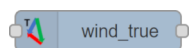
Name	Units	Description
<b>Yaw</b>	(°)	Heading (magnetic, 0/360)
<b>Pitch</b>	(°)	Negative with boat goes down (-180/180)
<b>Roll</b>	(°)	Negative with boat goes sartboard (-180/180)
<b>YawRate</b>	(°/s)	Positive clockwise
<b>PitchRate</b>	(°/s)	Negative with boat goes down
<b>RollRate</b>	(°/s)	Negative with boat goes startboard
<b>AWA</b>	(°)	Apparent wind angle, positive with wind from startboard (-180/180)
<b>AWS</b>	(Kn)	Apparent wind speed

Name	Units	Description
<b>TWA</b>	(°)	True wind angle, positive with wind from starboard (-180/180)
<b>TWS</b>	(Kn)	True wind speed
<b>SOW</b>	(Kn)	Boat speed (pilot)
<b>Declination</b>	(°)	

## Outputs

- **Serial out** : Prepare 25Hz frames to sent to NKE system

### 11.97 wind\_true



## Description



**FIG. 94**

Calculate the original true wind (TWA, TWS) in the horizontal plane using corrected measured wind and navigation data. Calculate TWD if course is available. Only one straight forward calculation is available, no option are required.

## Properties

- **Name** : Manta box name
- **Ref. to Course** : The True Wind Angle is referenced to the Course axis. Unchecked : Referenced to the longitudinal boat axis

## **Inputs**

### **Input 0 : Corrected Measured Wind**



Value Name	Units	Description	Calculations using this input
<b>CMW_angle</b>	(°)	Corrected Measured wind angle	TWA, TWS and TWD (Output 0)
<b>CMW_speed</b>	(kn)	Corrected Measured wind speed	TWA, TWS and TWD (Output 0)

**Input 1**

Value Name	Units	Description	Calculations using this input
<b>Course</b>	(°T)	Course	TWD (output 0)
<b>Leeway</b>	(°)	Leeway Angle	TWA, TWS and TWD (Output 0)

**Input 2**

Value Name	Units	Description	Calculations using this input
<b>BoatSpeed</b>	(kn)	BoatSpeed relative to the surface of water	TWA, TWS and TWD (Output 0)

**Output**

Name	Units	Description
<b>TWA_orig</b>	(°)	Original true wind angle (referenced on the horizontal plane passing through the MHU position)
<b>TWS_orig</b>	(kn)	Original true wind speed (referenced on the horizontal plane passing through the MHU position)
<b>TWD_orig</b>	(°T)	Original true wind direction (referenced on the horizontal plane passing through the MHU position)

**Notes****True Wind definition :**

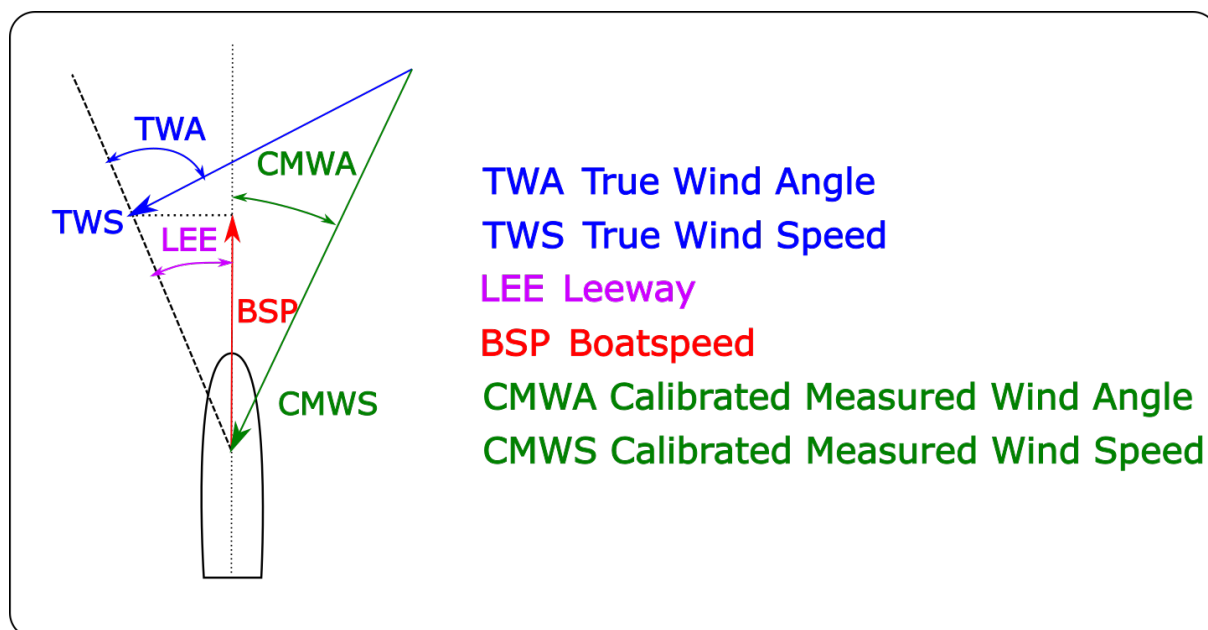
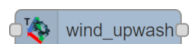


FIG. 95

## 11.98 wind\_upwash



### Description



FIG. 96

Use correction tables to compensate the True Wind from various symmetrical errors (identical on port and starboard tack) :

- Downwind Upwash : Wind perturbations at the Mast Head Unit position from the tip vortices (mast profile, the MHU wand, spinnaker, mainsail...)
- Mast attitude : The mast twist is function of reefing, running backstay tension...

- Residual errors : Not perfect sensors calibration

## Properties

- **Name** : Manta box name
- **Bypass** : If checked, no upwash correction is applied
- **Ext control** : If checked, add inputs to dynamically manage the correction tables
- **Save** : If checked, save dynamic update on the file. It is strongly recommended to not periodically send a dynamic table if this option is enabled.
- **Interpolation** : Select desired interpolation method
- **File format** : Select used file format
- **List of correction tables** :
  - **Set name** : Enter a name for the set of correction tables
  - **File TWA** : Select a true wind angle correction table (format, see note below)
  - **File TWS** : Select a true wind speed correction table (format, see note below)

## Inputs

### Input 0 : True Wind

Name	Units	Description
<b>TWA_orig</b>	(°)	Original true wind angle
<b>TWS_orig</b>	(kn)	Original true wind speed
<b>TWD_orig</b>	(°T)	Original true wind direction

### Input 1/2 : Ext correction table

Name	Type	Description
<b>CorrecTable</b>	FloatTable	Dynamic correction table. Variable name is free.

### Input 3 : Ext table selection

Name	Type	Description
<b>TableSelector</b>	Scalar/String	Dynamic table selector. Variable name is free. Use « Set name » or table set number as value. Number of table set starts at 0.

## Outputs

### Output 0 : Corrected True Wind

Name	Units	Description
<b>TWA_upwash</b>	(°)	Upwash corrected true wind angle (referenced on the horizontal plane passing through the MHU position)
<b>TWS_upwash</b>	(kn)	Upwash corrected true wind speed (referenced on the horizontal plane passing through the MHU position)
<b>TWD_upwash</b>	(°T)	Upwash corrected true wind direction (referenced on the horizontal plane passing through the MHU position)

### Output 1 : Status

Name	Units	Description
<b>AngleCorrection</b>	(°)	Current TWA and TWD correction
<b>SpeedCorrection</b>	(kn)	Current TWS correction
<b>TableSetSelected</b>		Number of selected table set
<b>TableSetSelectedStr</b>		Name of selected table set

## Notes

### Tables requirements :

- B&G WTP/Deckman, B&G H5000 or NKE Processor Regatta compatible format
- TWA must be > 0° and <= 180°
- TWS must be >= 0kn and <= 100kn

- TWA corrections must be  $\geq -90^\circ$  and  $\leq 90^\circ$
- TWS corrections must be  $\geq -90\text{kn}$  and  $\leq 90\text{kn}$

### Calculations above boundaries :

TWA correction (adjwa.d) :

- The spline surface interpolation is constrained in order to have a null correction at  $\text{TWA}=0^\circ$  and  $\text{TWA}=180^\circ$
- For TWS below the first TWS : the same TWA correction as the first TWS is applied
- For TWS above the last TWS : the same TWA correction as the last TWS is applied

TWS correction (adjvt.d) :

- The spline surface interpolation is constrained in order to have a null correction at  $\text{TWA}=0^\circ$
- The spline surface interpolation is constrained in order to continue with the same correction at  $\text{TWA}=180^\circ$
- For TWS below the first TWS : the same TWS correction as the first TWS is applied
- For TWS above the last TWS : the same TWS correction as the last TWS is applied

### Spline surface limitations and multiple upwash tables :

- The 3D spline interpolation can give bumps if 2 different corrections are too close and if the TWS/TWA are poorly distributed.
- The spline surface is smoother with only few corrections per TWS.
- If you require a step in correction function of TWA at angles that you need to change sails/reefs or if you can't rise the optimal sail set, you can prepare multiple upwash tables and select a specific sail set.

### B&G WTP/Deckman & NKE Processor Regatta file format

Size : maximum 6 pairs of Co/TWA, maximum 20 TWS lines.

	v1	a1	v2	a2	v3	a3	v4	a4	v5	a5
TWS	Co	TWA	Co	TWA	Co	TWA	Co	TWA	Co	TWA
TWS	Co	TWA	Co	TWA	Co	TWA	Co	TWA	Co	TWA

### B&G H5000 file format

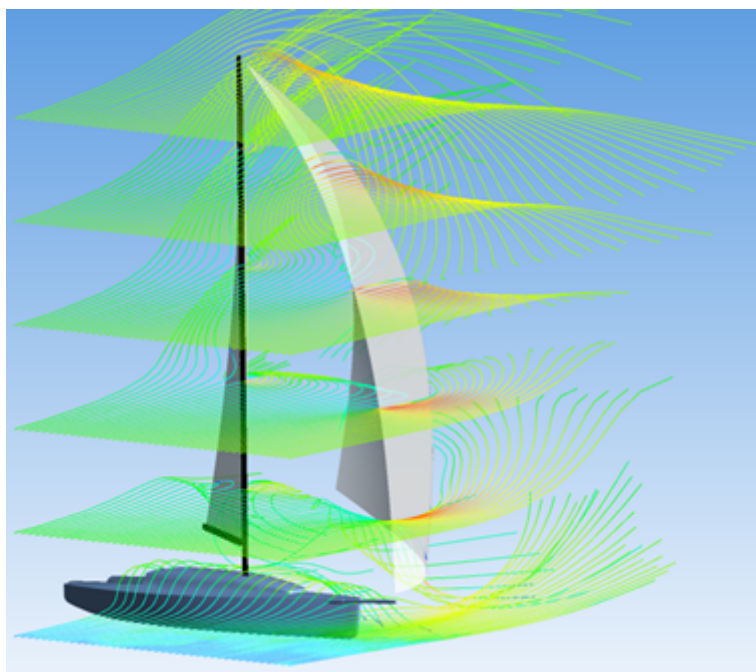
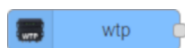
Size : 7 TWS columns, maximum 6 TWA lines.

	TWS	TWS	TWS	TWS	TWS	TWS	TWS
TWA	Co	Co	Co	Co	Co	Co	Co
TWA	Co	Co	Co	Co	Co	Co	Co

**Upwash modelisation :**

In a regular wind, the Ground Wind Speed and Direction must be similar on average when going from tack to tack or from upwind to downwind. This error is name *wiggle*. The angle and speed wiggle will be used to improve the upwash tables calibration (see the calibration help manual). You can use the True Wind Speed and Direction if the current is negligible.

The upwash and mast twist is physically applied on the apparent wind, we prefer calibrate their errors on the true wind and back calculate the apparent wind in order to make a mental calculation calibration possible.

**FIG. 97****11.99 wtp****Description**

Import B&G WTP3 sailing instrument variables.

## Properties

- **Name** : Manta box name
- **Multicast IP address** : Enter UDP multicast IP address
- **Multicast port** : Enter the UDP multicast port number

Use the variable list to check/uncheck the WTP3 variables to export. *Select All* and *Unselect All* buttons can be used to select/unselect all the variables.

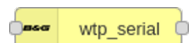
## Outputs

- The list of selected WTP variables

## Notes

Output Frequency : 10 Hz Variables must be declared in the *bg\_vars.d*, where the EOL symbol is « CRLF ».

### 11.100 wtpSerialManager



## Description

Ease of use WTP3 Serial interface : \* Configure WTP3 Serial module \* Prepare data to send through N2000 \* Extract received data from N2000

## Properties

- **Name** : Manta box name
- **CAN port** : Select CAN port
- **Device selection** : Select WTP Serial module
- **Instance** : WTP Serial module instance
- **Port** : WTP Serial module connector (COM1 = connector A)
- **Direction** : WTP Serial module usage
- **Baudrate** : Serial communication baudrate
- **Data bits** : Number of data bits to use

- **Stop bits** : Number of stop bits to use
- **Parity** : Parity configuration to use
- **Protocol** : Serial communication protocol to use (according to WTP Serial module wiring)
- **Output mode** : Output mode to use

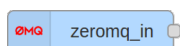
## Inputs

- **PGN 65280** : Must be connected to port 65280 of a net2000 in box with type set as WTP3 Serial
- **PGN 130822** : Must be connected to port 130822 of a net2000 in box with type set as WTP3 Serial
- **PGN 130823** : Must be connected to port 130823 of a net2000 in box with type set as WTP3 Serial
- **Data** (*opaque*) : Serial data to send (only if module is used as output).

## Outputs

- **PGN 65280** : Must be connected to port 65280 of a net2000 out box with type set as WTP3 Serial
- **PGN 130822** : Must be connected to port 130822 of a net2000 out box with type set as WTP3 Serial
- **PGN 130823** : Must be connected to port 130823 of a net2000 out box with type set as WTP3 Serial
- **Received data** (*opaque*) : Received serial data (only if module is used as input).

### 11.101 zeromq\_in



## Description

Get data from a ZeroMQ interface.

## Properties

- **Name** : Manta box name
- **Client** : Check to work as a client, otherwise it works as a server.
- **Local port** : Only for server, enter the local port to receive external connection
- **Remote IP** : Only for client, enter the remote IP address to connect to
- **Remote port** : Only for client, enter the remote port to connect to

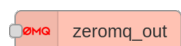


- **Type** : Select the type corresponding to the desired communication model (Req/Rep, Pub/Sub, Push/Pull...)
- **Topic** : Only for *Subscribe* type, enter an optionnal *topic* to accept only incoming messages beginning with this specified prefix
- **Add TS** : Add timestamp to output data flow

## Outputs

- **data** (*Opaque*) : ZeroMQ received data

### 11.102 zeromq\_out



## Description

Send data to a ZeroMQ interface.

## Properties

- **Name** : Manta box name
- **Client** : Check to work as a client, otherwise it works as a server.
- **Local port** : Only for server, enter the local port to receive external connection
- **Remote IP** : Only for client, enter the remote IP address to connect to
- **Remote port** : Only for client, enter the remote port to connect to
- **Type** : Select the type corresponding to the desired communication model (Req/Rep, Pub/Sub, Push/Pull...)

## Inputs

- Data to send. Must be of *Opaque* or *String* type. Note that string variables are limited to 32 characters.

## 12 Appendix

### 12.1 Récupérer l'adresse IP de l'Exocet

Dans le cas où l'adresse IP de l'Exocet aurait été oubliée, une adresse IP de récupération est disponible. Pour cela, connectez directement l'Exocet à un ordinateur. Fixez l'adresse IP de l'ordinateur à 10.0.0.2 (masque : 255.255.255.0)

Ensuite, sur un navigateur web, connectez-vous à 10.0.0.1. L'interface web de l'Exocet est maintenant accessible. Changez l'adresse IP commune pour celle que vous désirez.

### 12.2 Configuration de votre PC en tant que serveur de temps

Consultez le support Microsoft à l'adresse :

<https://support.microsoft.com/en-us/help/816042/how-to-configure-an-authoritative-time-server-in-windows-server>