## Implementation in the sensorlist

### Introduction

As the sensorlist is way bigger and more complex than the devicelist, we will not fill in all the columns and fields. We will give some excerpts from what you can expect at the different devices and the different columns. On the basis of the single line drawing we used for the explanation of the devicelist, we will give as many examples as possible. After this explanation, you should be capable to work out the rest of the sensorlist.

*: Once you have imported the sensorlist into NavVision, most of the fields will be automatically added. This will be done by NavVision on standard basis. This will not always be right, so you need to check that. We will come back at that in a separate Chapter*

### Import Result

The import result is a checklist. When you have imported the sensorlist, NavVision will generate a few files at which we will come back later. One of these files is the sensorlist\_generated. In this file you will see in the first column the import result. For more information we refer you to Chapter 12.5.

### ID, CableLabel, GroupLabel

These columns are optional. They are not needed for the proper functioning of the program. However it could come in handy when you fill up some of those fields.

The ID column you can use for your own reference. Maybe you use some kind of numbering that is different from the one you get from the shipyard.

Many installation companies use cable labels (numbers) for the connections of the wires at the terminals and/or at the sensor/control. If you fill in these Cable labels in this column, you will have a reference in the sensorlist which is searchable. You also get a reference in NavVision where the Cablelabel is shown in the Wago screen.

In the GroupLabel column you can separate different (alarm)groups and their I/O. This makes it quite easy to search specific I/O or just select a whole group that you need to adjust.

Next figure will show a small example:

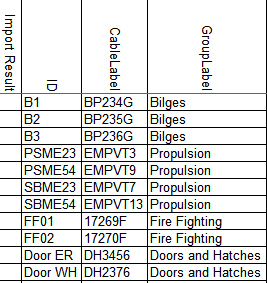


Figure 11‑1: ID, CableLabel, GroupLabel Example

### Item

The Item is somewhat different and needs some attention. In consultation with the installer or even with the shipyard, you need to come up with a descriptive name for each field (I/O, sensor, control). As this is the name that comes up in the logbook and the alarmlist, you need to be clear about what it is.

Sometimes people come up with texts like “Preferential Trip & Em. Stop System Power Failure”. As you can see it is quite long and also very confusing. It can mean a lot of things. Maybe this one would be easier to understand if you called it “PMS Power Failure”. It is certainly more descriptive and short and concise.

In other cases, the crew can be very familiar with certain names. The example “N.16 Fr 20-21 Bilge Level High Alarm” may seem confusing, but the crew knows exactly what it means cause they have been working with this name for years.

Remember however that the text is free to choose, but it will appear in alarm lists and the logbook. So keep it as simple as possible.

#### Conjunction with SensorType

You also need to understand the conjunction with the “Item” column and the “SensorType” column. As explained in Chapter 11.3.5 SensorType defines which subfield or action of the Data Field is set by the value in that column. So if it is not “standard” you better check the “Item” text again.

For example: A sensortype can be “High Alarm” or “Running” or even just “Alarm”. This means that you trigger an extra action with the sensortype field. Now let’s say that you have the sensortype defined as Alarm. When you put “Bilge ER Alarm” as text in the “Item” field you get it double. With an alarm you now will get “Ext: Bilge ER Alarm Alarm” in your alarm screen. Easy to understand that if you use the sensortype “Alarm” you leave the word Alarm out of the Item-text. This is valid for all the conjunctions between these two columns.

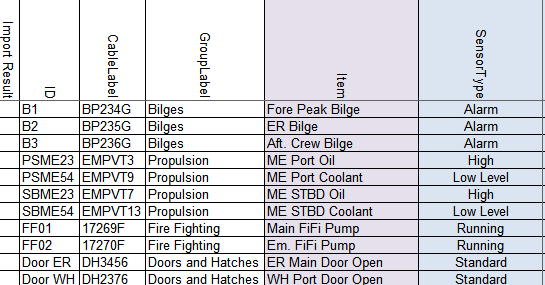


Figure 11‑2: Item example

### SensorType

SensorType defines which subfield or action of the Data Field is set by this value. By default it’s “Standard”. Standard means it’s not defining a subfield or action, but the value of the Data Field itself. (For more options see Table 11‑2 and Table 11‑3).

With “standard” as option in the sensortype column NavVision will only act upon the field itself. So if the field is an alarmfield NavVision will give an alarm when that field gets triggered. This goes for all the different type of fields. So if for example it is a Pressure field (analog value) NavVision will show the pressure value. If you don’t fill in anything in the sensortype column, it will automatically be “Standard”.

If no extra action is necessary on a field you probably won’t use the sensortype column. This comes in play when you want something extra. An analog field that needs a “high” alarm. An output that needs a “Set” request and so on. Before we elaborate on this we need to explain something about the “Fields” within NavVision.

#### Fields

NavVision works with a database with all kind of ID’s in it. These ID’s are represented by fields that are divided into sup-parts. Every action in NavVision revolves around this database of field-id’s. You can use one field over and over again cause the main value is set in the database.

Once you connect a sensor or control to a field you can do almost everything you like. For example if you want to control a pump with a hardwired button, you can connect that pump in NavVision to let’s say the field “Pump1”. Through a Wago PLC you now get to control that pump. On a Wago Digital Input, you connect the field “Pump1” and you hardwire a button to the same Slice. Now if you push the button the Wago input will get high. If you put the same field “Pump1” to an output on the Wago. This output will get active as soon as the input gets active. While this is an output, you can hardwire it to the actual Pump1. So than when you push the button the pump will start to run.

These fields you can find in the “fieldlist.txt”. Once that NavVision is started for the first time, you will find it in the root folder. You can open and control this .txt-file best with Excel. For people not familiar with Excel there is a small explanation in Chapter 9.2.

As there is a lot of intelligence in the fields already it is good to understand the interaction between the field and the sensortype. You can mess things up when you use this wrong.

#### Back to SensorType

So, as mentioned earlier, there is a conjunction between the “Item” and the “SensorType” and now also between the “Field” and the “SensorType”. We use the same example as in Chapter 11.3.4.1 to show how it all fits together.

As we mentioned in that chapter, you need to pay attention to the name you use in the Item-column so you do not get confusing or double values. Same goes for the fields and the sensortype. If you choose a field that is already an alarm-field this means that, when the value gets high, the field will give an alarm. So it is not necessary to put an extra alarm in the sensortype column. This is not only double but also can confuse the system or the user. On the other hand, if you use a field that holds Level information, you might want to trigger an alarm when you get to a certain level. This is possible by putting “High Level” in the SensorType column. You see there is quite some interaction between those different columns.

You need to practice a lot with the sensorlist to learn how to work with it. For now we will give an example on how it is not supposed to be concerning “Item” “SensorType” and “Field”.

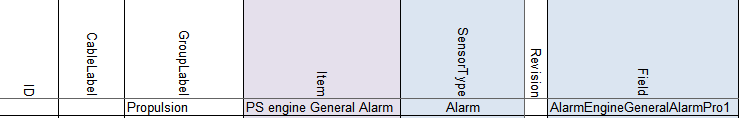


Figure 11‑3: Double fault

As you can see we have an alarmfield in the field column, a sensortype that triggers an alarm and the name in the Item column that will make it double. Easiest in this case is: keep the alarmfield in the Field column, put Sensortype to “Standard” and take “Alarm” out of the Item column name.

### Connection

Connection defines the type of connection for digital in- and outputs. Connection is NO by default. If an in- or output is normally closed it’s NC. If you have problems with switches that go the wrong way around or there is an alarm where the sensor itself is not in alarm, this is the first place to look.

### Device

Identification of the device where the sensor/control or serial device is connected to. This text should be unique for each FT NavVision® device. The text is case sensitive.

This device is already been set in the devicelist. See chapter 10.5.2 to see how you’ve done that. Now all the I/O that you put into the sensorlist must be connected to the right device, so NavVision knows where to look for it and how to process it.

When you click on a field you can see there is a drop-down menu. In the menu you will find all the previous assigned devices. All you have to do now is choose the right device.

As we look at the single line drawing and we take the example we had earlier we can tell that the Fore Peak Bilge is connected to the Wago AC Room and the ER Bilge is connected to the Wago 2 ER.

The Port Engine is connected to the second port at the second SerialLan in the ER so you choose Serial 2 ER-2 as device. It will look a bit like the following figure:

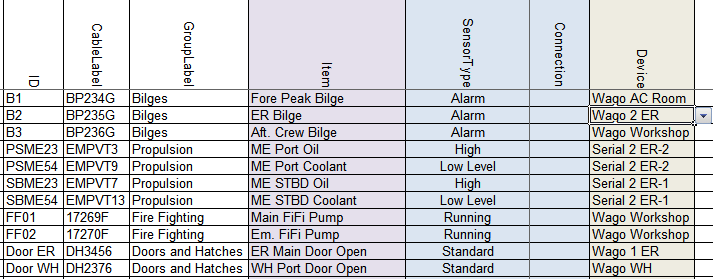


Figure 11‑4: Sensorlist device column

Of course, while you probably will start filling all the I/O’s from one device at the time, you will get a long row with only Wago WH and then for example Wago Workshop. You will see that once you start working with it.

In the example we only have serial and Wago connections, but it can be anything that you filled in as a device. It is probably best to start to fill the list with the Wago devices as these are mostly already assigned. Later on you take the serial connections with for example Modbus or CANbus on it.

### Location

For location you can use the same field as in the devicelist. It is optional, but also usable for sorting the list and/or localizing sensors or I/O’s.

### Interface

Here you define what kind of interface is used to connect the sensor/control to NavVision. For Wago this is divided in the slice’s type-number. For Modbus, Canbus and other protocols it is Serial (Digital/Analog) in or out.

If you have the Wago drawings available, it is easy to choose the right module for that. If you have trouble finding it, you can always fall back to the documentation of Wago. For the protocols you just need to look if it is a digital or analog value and if it is an input or an output. More on these serial interfaces we discuss later.

To give you an idea, we go back to our example. The bilges in the example will be most likely digital inputs. As Wago works standard with 24V it will be a Dig in (24V) you have to choose there. This goes also for the fire pumps and the doors. Probably normal switches so an input of 24V for High or Low (On or Off).

The engine however is somewhat different. As we can see in the SensorType field it is just a digital input where On is High or On is Low Level. However, this is the SensorType Field. This field will give NavVision a reason to calculate an alarm on an analog value. So don’t be misled. This will be an analog field coming in (Oil is a pressure field and Low Level is a level field). So you will have to connect it to an analog interface module on the Wago. This can be 4-20mA, 0-10V or a lot of other sorts. Let’s say the oil pressure field is a 4-20mA signal and the level field is a 0-10V signal. We will come to the following:

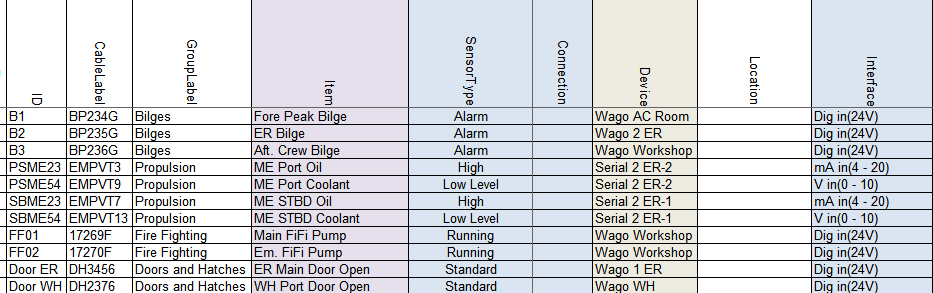


Figure 11‑5: Sensorlist Interface column

### Module

For Wago you start counting the slices after the 750-626 module. Starting with 1 and so on. If you do not filter the sensorlist, than it will be hard to look if the numbers are alright. But as we will explain that in a later stadium we now just have to watch carefully. As example we show you the next figure:

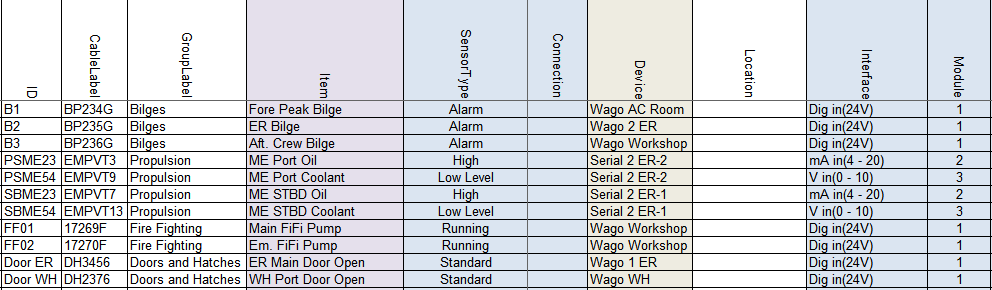


Figure 11‑6: Sensorlist Module column 1

This may look a bit odd, but realize that we put the Dig in (24V) on a module with 8 contacts (Pin). So the first 8 DI you find are on the first module. Same goes for the mA in (4-20). These modules have 4 contacts. It will become more clear in the next paragraph.

For CANbus in this column you put the PGN or Parameter Group Number as index for the I/O. With Modbus you take the Modbus mapping as starting-point. The register of the Modbus mapping you put here. See following example:



Figure 11‑7: Sensorlist Module column 2

### Pin

The I/O index on the module for WAGO and the bit offset in the message for serial protocols. (NOTE: The pin index is 1 based).

#### Wago

So if you look at a Wago slice you will see openings for the wires to be attached. It needs some attention because Wago has a different numbering than NavVision and this can be confusing. First let’s look at the numbering Wago uses:

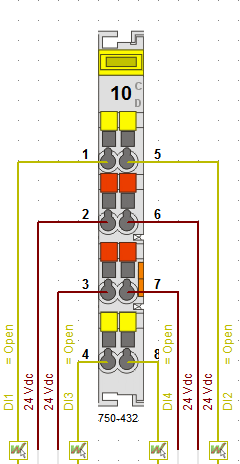


Figure 11‑8: Wago Numbering 1

As you can see Wago numbers the pins vertically so left side 1-4 and right side 5-8.

NavVision has to number different because of program issues. We number the Wago horizontally. So 1=1, 5=2 and so on. You have to keep that in mind to work properly with the sensorlist. The NavVision numbering will look as follows:

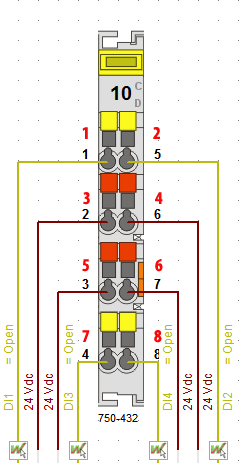


Figure 11‑9: Wago Numbering 2

So when you number it this way in the sensorlist, it will mainly look like the following figure:

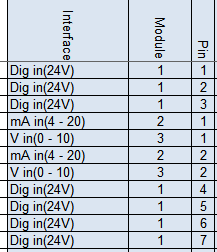


Figure 11‑10: Pin column 1

Or, when you already sorted the sensorlist, it will make it even clearer. See the following figure:

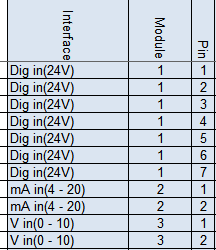


Figure 11‑11: pin column 2

Of course I can’t show you the example from where we started off. While all the connections where on different Wago’s there, we should than have divided all the modules over the different Wago stations. Therefore, before you begin numbering the modules and pins, you need to have all the Wago connections in the sensorlist. Then you can filter the sensorlist first (as explained in Chapter 11.4) and then do the modules and pins.

#### Serial Protocols

For Serial protocols the pin number defines the bit-offset. So if you need to connect to a serial protocol at bit level, this column is where you assign this. Note that the “pin index” is 1 based. So if you need bit 3 for PGN 65280 you have to put 4 in the column. See next figure:



Figure 11‑12: Pin column 3

### Type

#### Wago

For Wago you fill in here the module number. You can find the module number on the Wago slice itself, on the drawings or look it up in the Wago documentation. See following figure:

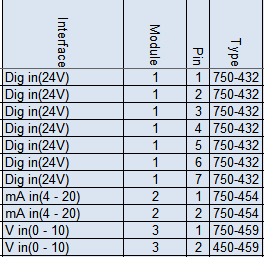


Figure 11‑13: Type column 1

#### Serial Protocols

For CANbus we do not use this column. For Modbus we use this column to define the function code of the Modbus register. So for example if you read actual values in Modbus, this will be Modbus function 04. Type a 4 in the “Type” column. See following figure:



Figure 11‑14: Type column 2

### Min-Max

The columns Min and Max show the range of the data field. This will come back in instruments and value-bars. If you choose them wrong then you get values that go beyond the range of an instrument. Once you see this, you know that you need to change the values. If you get the right data from the shipyard you can fill it in in these fields. For digital data it is Min=0 and Max=1. It is not necessary to fill in the Min- Max-values for digital values. NavVision will do this for you. You can also change these values at a later time.

### DefaultUnit

The defaultUnit is used to set the unit to present this Data Field in. This can also be changed in the instrument or mimic itself, but for big amounts of data it is easier to use the sensorlist. If you do not choose anything NavVision will fill it in for you. For options see the next figure:

|  |  |  |  |
| --- | --- | --- | --- |
| Alarm | High Alarm | Low Alarm | Ampere-Hour |
| Radians | Degrees | Grads | °/sec² |
| rad/sec | °/sec | °/min | Normal |
| Normal | Mirror | Liter | Gallon |
| GallonUK | Cubic Meter | Percentage | L/km |
| G/Nm | L/min | L/Nm | G/S |
| l/h | G/H | Guk/H | G/min |
| L/S | Guk/min | l/m | Guk/S |
| Count | Degrees | Grads | Radians |
| Kilo Ampere | MilliAmpere | Ampere | Dampening |
| kg/m³ | kg/L | lb/gal | nm/G |
| nm/l | m/l | km/l | Poundal |
| Newton | Lbf | Kgf | Kips |
| Newton Meter | Kgm | Lbf-ft | Hertz |
| m/g | nm/kg | Km/Kg | kWh/L |
| kWh/Guk | kWh/G | Fathom | Nautical Mile |
| Feet | mi | cm | Km |
| mm | Inch | M | cd/m² |
| Kg/H | g/s | t/s | Name |
| Okta | Mask | Number | Percentage |
| Degrees | Bar | mBar | kPa |
| Hg | hPa | MPa | Psi |
| Pascal | MilliOhm | Ohm | KiloOhm |
| RPM | Hertz | RPM/s | Km/H |
| Knots | M/Min | M/S | Beaufort |
| Miles per hour | Feet/Min | g-force | m/s² |
| OnOff | Open | Alarm Group | General Alarm |
| Switch Off | Alarm Deadman Group | Switch | Take Over |
| Push | Popup Switch | Kelvin | Fahrenheit |
| Celsius | Date | Day | Date & Time |
| Month | Date & Time Left | Time | Sec |
| Week | Hour | Year | Min |
| mSec | uSec | MilliVolt | KiloVolt |
| Volt | Volt Ampere | kVA | Volt Ampere Hour |
| kVAh | MVAh | Watt | MegaWatt |
| KiloWatt | MegaWattHour | WattHour | kWh |
| Ton | kg | Lbs | Gram |

Table 11‑5: (Default) Unit options

For our example it will be the following:

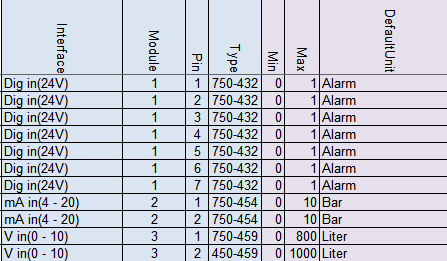


Figure 11‑15: Default Unit column

### Manufacturer

This is an optional field for your own convenience

### Supplier

This is an optional field for your own convenience

### Comment

This is an optional field for your own convenience

### Revision

This is an optional field where you can give a revision number. Easy if you need to see when something has been changed or what has been changed after a certain revision.

### Field

This is one of the most important columns within the sensorlist. This is the place where you assign a dedicated field from the database of NavVision. This field will be inextricably linked to that I/O, sensor or control. All the in- and outputs and all the calculations, as well as connection to instruments and mimics, will be represented with that field. Also the alarmgroup and behavior will be defined by what you choose here.

You can understand that it is utmost important that this field is chosen properly and a field is only used for one particular sensor/control. These field-column is also the one that will consume most of the time in building the sensorlist.

As mentioned before these fields can be found in the file “fieldlist.txt” in the root of NavVision after the first start of NavVision. Also FTSelect is suitable for checking the right fields.

#### How to work with fieldlist.txt

To find all the right fields you first have to open “fieldlist.txt” the right way. You need to know that, to work with the .txt-file you need to open it in Excel. To do so, right-click on the .txt file and choose “open with” and go for Excel (see Figure 11‑16)

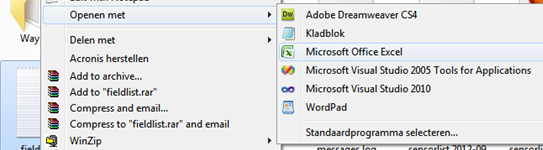


Figure 11‑16: open with Excel

Now the program will open as an Excel sheet, with all the opportunities. There are two things you must do first (this is basic Excel knowledge).

Click in the upper left corner of the sheet (see Figure 11‑17) to select all fields. Put your mouse between row “A” and row “B” (see Figure 11‑18)and doubleclick. The fields now will be all on the right width.

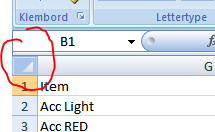


Figure 11‑17: Excel 1

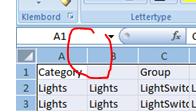


Figure 11‑18: Excel 2

Now select the first row by clicking with you mouse on the number “1” in front of the row. Goto Start>sort and filter and then filter (see Figure 11‑19). Click it

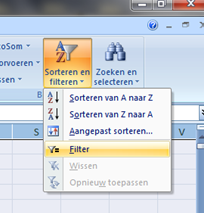


Figure 11‑19: Excel 3

The first row with the index names has now drop down menus and you can choose what to filter. For our example we need Bilges. Goto the index name “Category” click on the dropdown menu, deselect the “select all” checkmark and then select the “bilges” checkmark (see Figure 11‑20). You now have only all the bilges-fields available.

You can narrow it down by going to the index name “Group” and make another selection (see Figure 11‑21). In our case it is AlarmBilge

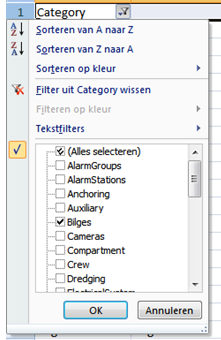


Figure 11‑20 : Excel 4

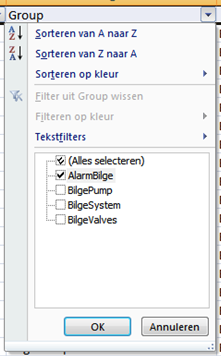


Figure 11‑21: Excel 5

Now we’ve done this we have only the Bilge alarmfields available. You can figure out yourself how you can further narrow it down, or use it for other fields.

#### How to work with FTSelect

In the bin-folder of NavVision you will find the program “FTSelect”. Open it and you will get the following window:

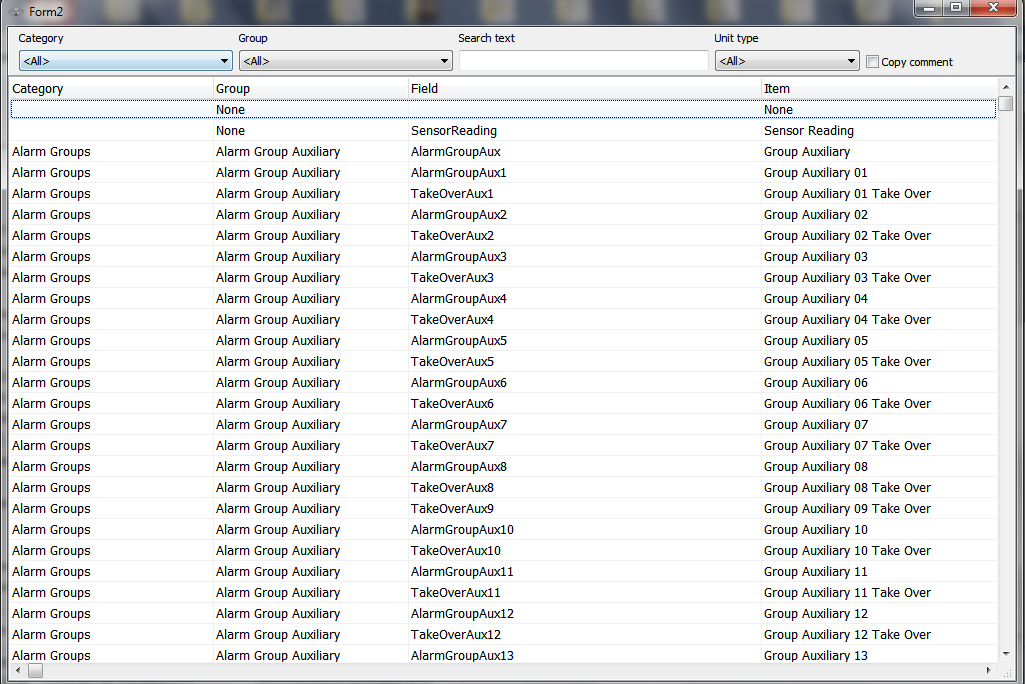


Figure 11‑22: FTSelect

You can search for a specific text, or just choose a category and a group to get to the right field. You can even filter on “Unit Type”.

Say you look for the bilge alarm, you can fill in the fields as follows:



Figure 11‑23: filtering in FTSelect

You now get all the bilge alarms available as seen in the following figure:

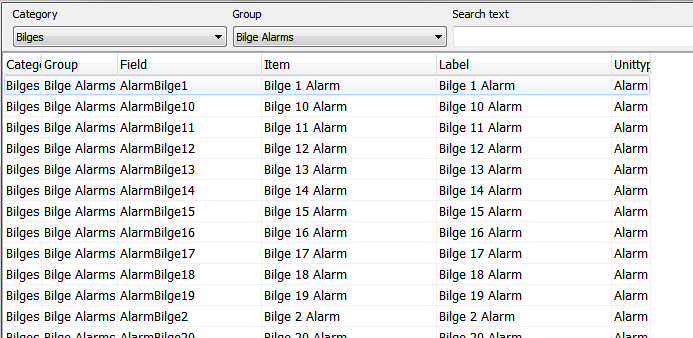


Figure 11‑24: Filtered FTSelect

Now you just select the right line that you need and highlight it by clicking on the light. FTSelect will automatically copy the field to the clipboard and you can paste it again in the sensorlist.

#### Back to the Field column

So now we have narrowed it down to the right fields, it is time to give all our I/O a separate field tag. In the adjusted fieldlist.txt we now see all the alarms for bilges available. We need three bilge alarms, so we need three distinctive bilge alarm fields.

In the previous mentioned excel list, goto the column “Field”. As we are just starting, all the fields are still available. So we can choose the first three Bilge Alarm Fields. Select these three fields and copy them (CTRL-C). Go back to your sensorlist and past them into the field column behind the three bilge items. See the following figure.

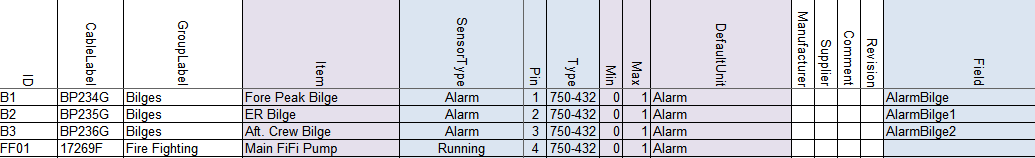


Figure 11‑25: Field column 1

You can follow this for all the other fields and you will get the following:

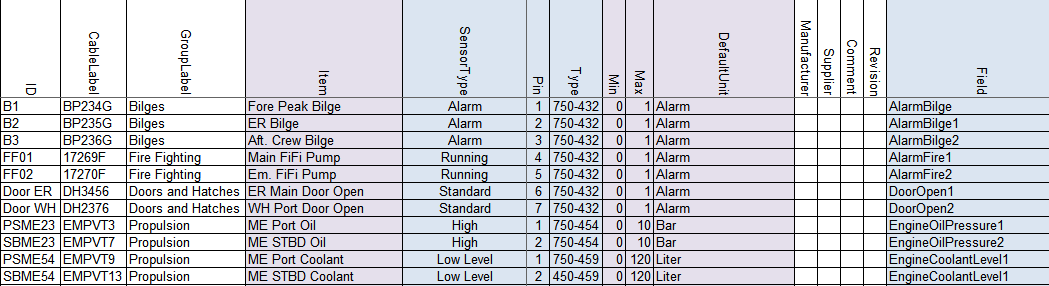


Figure 11‑26: Field column 2

*: with bigger projects it is easy to get mistaken. Easiest way to prevent this is that you color the fields u have used in the fieldlist.txt yellow. That way you will know which ones are used and which are free. Later on we show you that NavVision has a way to trace the faults. See chapter 12.5*

### Label

The Label column exists of the short description of the Data Field when shown in an instrument. Default label text belonging to the Data Field is preferred.

So the easiest way is to copy the “Item” column and just past it into the “Label” column. This way you have a one-on-one connection. Off course this is the text that you find as a label in instruments etc. When the text is too big, it won’t fit in the instrument or just looks sloppy. So if this is the case, just alter the name here to a short description. “Emergency Generator Power Failure” can be changed into “Em. Gen. Power Fail.” And if the default unit is available in an instrument, you can even leave types as “Pressure”, “Voltage”, etc. out of it, cause they will see that it is a “Bar” value or a “V” value. So “Main Engine Lub Oil Pressure” can be set as “ME Oil”

### Rate

Rate describes the number of samples per second of a sensor/control. This is defined by the protocol. Leave empty.

### Index

Index defines when this Data Field Definition [DFD] is valid. The Index column can only be used in combination with a Data Field Definition [DFD] that has the SensorType set to Index and is in the same message as this DFD. Default is empty.

### Datatype

DataType is used to define the type of value on serial protocols. For analogue values it’s Float, Signed or Unsigned. For digital values it’s Bool. For enumerations this is Enum. See Enum column.

### Enum

Enum is the index value where the received value should compare to, to switch the Data Field on. If the value is not equal to the Enum index the Data Field is switched off.

### Count

Count is the number of bits starting from the pin index. For a digital value it’s typically 1 with a pin index between 1 and 16 and for analog values it’s for example for Mod bus typically 16 with pin index 1. Canbus has almost always a count of “2”.

### Multiplier

Multiplier defines the factor between the sensor/control value and the real value.

For inputs/read:

*value = sensor value \* multiplier + offset*

For outputs/write:

*sensor value = (value – offset) / multiplier*

For example: if the temperature is send in from a sensor in whole numbers (210 for 21 degrees) you can put in a multiplier of 0.1. So when the sensor sends 210, it goes through the multiplier and NavVision makes it 210\*0.1=21

### Offset

Offset defines the offset between the sensor/control value and the real value. See Multiplier column.

### Unit

The Unit in which the sensor/control value is received or send. Directly from the sensor control. This field differs from the DefaultUnit by the fact that NavVision has no influence on this one. For options see Table 11‑5.

### GroupLocal, ItemLocal and LabelLocal

To use a local language in NavVision it is possible to set local fields. In chapter 11.1.14 Software installation and commissioning manual 1.9, you can find how you can set the local language for an operator.

As you will set the English language in the columns Group, Item and Label, you can set the local language in these columns. For example you can set all these items in the Chinese language. Now by choosing the local setting for an operator in NavVision, this operator will get NavVision to show his preferred language

### Decimals

As described in chapter 11.2.2.4 Software installation and commissioning manual 1.9, you can set the number of decimals that you want NavVision to show in values. To change it quickly for a lot of values, you can use this column. Empty is standard NavVision settings. Other choices are 1,2,3,4,5,6,7 or 8

#### 

### Other columns

The rest of the columns in the sensorlist are optional, because NavVision will fill them in for you. These fields will only be used for specific needs. If you want to know what you can do with these columns, it is enough to look in the Sensorlist Table (see Table 11‑1).

## Filter sensorlist

Once you start filling the sensorlist it is good habit that before you fill in the columns module and pin, you filter the sensorlist. This is also common Excel knowledge, but for your convenience we will give a short explanation here.

Let’s say you have filled in a few I/O that you got from a list and you just start to fill in in no particular order. Than it is impossible to address the right module and pin as the list will be extremely long and changes on mistakes will be huge. So before you start with the module and pin columns you will have to filter the sheet.

The columns that you did fill in contain the device-column and the interface-column. With these two you can filter the sheet for a first result.

What you need to filter first is that all the devices are grouped and the interfaces are grouped together. To do this you go to Start>Sort and Filter>Custom sort. You will get a menu like in the folloing figure:

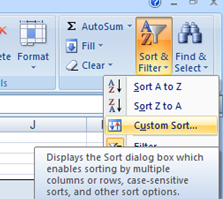


Figure 11‑27: Custom sort

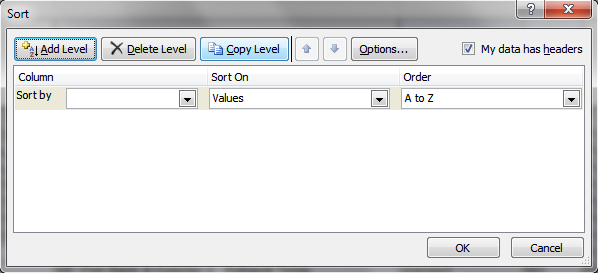


Figure 11‑28: Custom sort window

In this window you can add as many levels as you want to filter out the sheet. We need only two for now, “Device” and “Interface” as you see in the next figure:

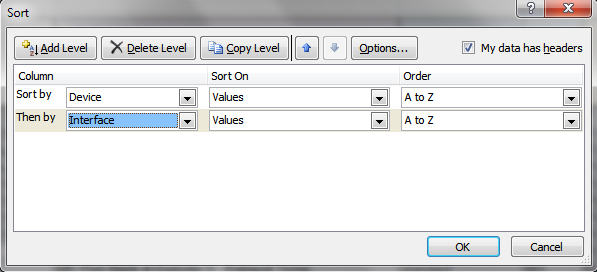


Figure 11‑29: Sorting device and interface

Sorting it this way gives you the devices ordered at the right Wago PLC and you get all the same slices together. This is the first step of filtering that is pretty easy and it gives the following example:

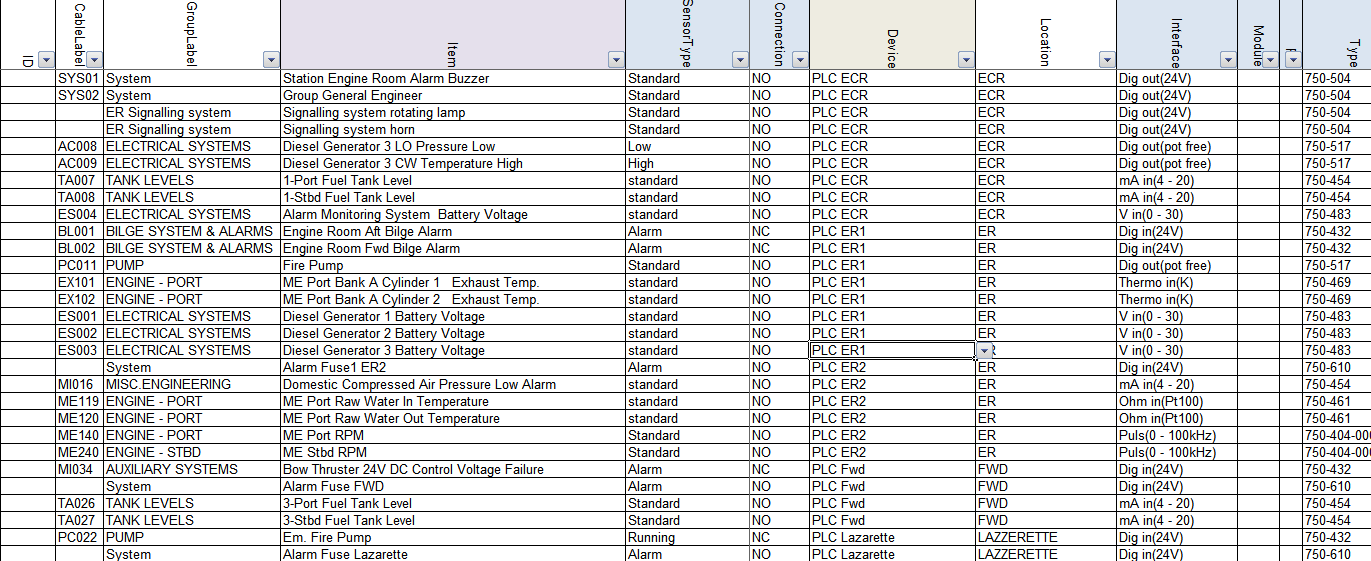


Figure 11‑30: Sorted sensorlist

As you can see we have all the devices put together and within these devices we have all the interfaces put together. Due to the running sequence Wago follows, we need to make some final adjustments by hand. There is a certain sequence that we have to build up the Wago PLC’s in. For more information we refer to Wago. For now we can say that we start the construction of Wago in the following (global) order:

First DI-modules

Than DO-modules

Than AI-modules

Than AO-modules

This is a global distribution, cause it sometimes needs some additional action. For now this is enough to understand.

As you look at Figure 11‑30 you can see in the column “Interface” that it worked out pretty well. The only thing in this example that is not right are the modules at line 28 and 29. This is need to know knowledge. These modules don’t work in that position and has to be places before the 750-454 module at line 25.

To do so select the two lines (28 and 29) and cut them.

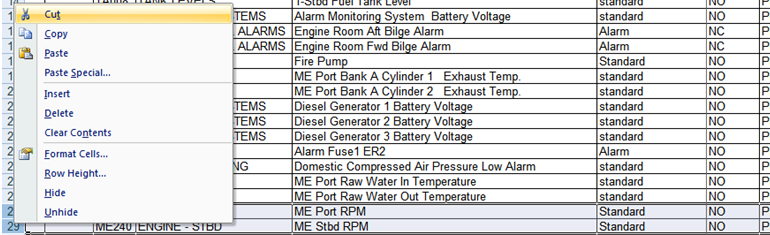


Figure 11‑31: Cut and paste 1

Once you’ve done that you go to the line that you need to insert them and right-click on the number of the row underneath that line. Choose Insert Cut Cells. See following figure”

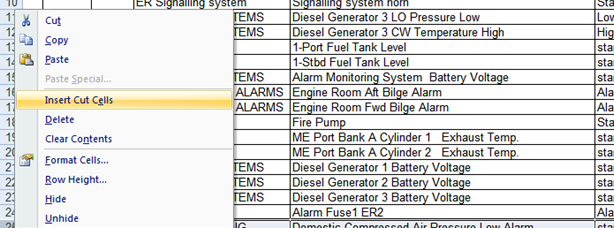


Figure 11‑32: Cut and paste 2

Now you have everything in the right order and you can start numbering the Modules and Pins.

*:You need to have good to excellent knowledge about Wago and Microsoft Excel. We recommend that you get some additional training on this as well.*

## Special issues

There are several special issues that you can put in the sensorlist. Changes you make in NavVision itself will get lost as soon as you import a new sensorlist. To prevent this loss it necessary that you put all the changes you make in NavVision are directly put into the sensorlist. In the hectics of a commissioning it will not always be possible to do that directly, for adjusting the sensorlist at a later time we refer you to Chapter 12.

However we do like to give an example of things you need to change by hand in the sensorlist. For this we assume that you have more than basic knowledge of working with NavVision.

So let’s say that you have a ship with a lot of duty-stations. At some point the crew will ask you to change the names in the alarm/duty mimic, so they can see who is on duty or who they are calling through the NavVision call function.

Given the next example (see Figure 11‑33 and Figure 11‑34) we have changed the names of a few files to match the names as the crew would like to see it. As you will know these names are changed in Fieldsettings>Comment>Crew>CrewAlarms within NavVision. If you do not put this in the sensorlist, each time you import a new sensorlist these names will be changed. This is not desirable, so you need to put these changes into the sensorlist.

If you put this in to the sensorlist, the easiest way to do this is on top of the list. Add some extra rows and start filling the information there. You have to understand that it is NavVision based so the device is NavVision NavVision. SensorType is Standard, Connection is NO and in the “Item” column you fill in the name that you want to show in the alarm mimic of NavVision (see Figure 11‑35).



Figure 11‑33: Duty names



Figure 11‑34: Call names

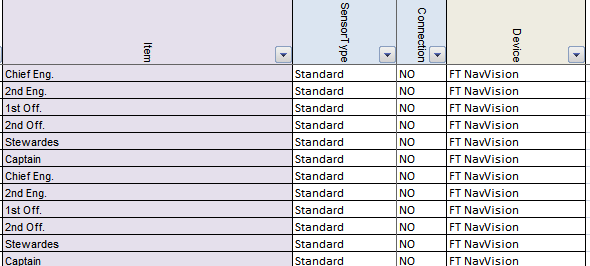


Figure 11‑35: Special issues 1

At the “field” column you assign the right fields (which you will find in the fieldlist.txt see Chapter 11.3.19.10). In the “Label” column you once again fill in the names as you described them in the “Item” column (see ).

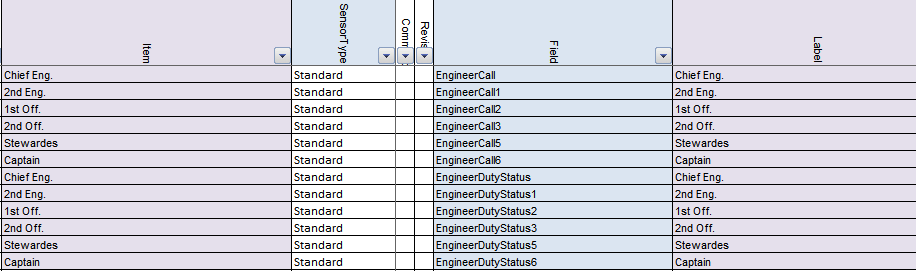


Figure 11‑36: Special issues 2

That is all. NavVision will take care of the rest. Now if you import the sensorlist again, you will keep the names you gave to the Crew Alarms.