



## The CMX Alternative Interface (AI)

The CumulusMX AI is an alternative to the supplied Interface for the system. It is intended to be used in addition to rather than as a replacement for the supplied interface. This is to ensure that you always have the access provided and developed by Mark and updated by him as required. Upgrades to the AI will be provided as soon as feasible when upgrades are developed by Mark but no guarantee is given or implied that the AI will provide all the features that Mark develops.

**This version is intended for CumulusMX version 3.26.0 build 3256. It will NOT work on earlier builds.**

I have still not been able to make the 'Settings ~ Internet Settings ~ Web/Upload Site ~ General Settings OR Advanced Settings' tabs fully expand. If these need adjusting you will have to use the default interface.

### Installation

The supplied zip file contains this document and a separate zip containing the AI files. Please read this document for information relating to the changes and functionality of the this version.

Please remember that this is **not** a replacement for the supplied interface. If you use it as a replacement, it will be lost when you next upgrade CumulusMX.

- Unzip the ai2 packed to its own folder not part of your CumulusMX installation.
- Copy the ai2 folder into the existing interface folder for CumulusMX making sure it's a subfolder of it; I.e. CumulusMX/interface/ai2.

**Warning** if you don't make it a sub-folder, you will corrupt the default interface; if you don't copy it to the interface folder, it simply will not work.

- You should now be able to access the AI using:
  - i. localhost:8998/ai2/
  - ii. <ip address>:8998/ai2/
- You may need to clear your cache to prevent old versions of the support files being used by mistake.



## *The folder structure is:*

- ai.cmx - The html files for this upgrade. I have changed the name of some of these files. This includes a 'development.html' page that you can use for experimentation and development!!
- /css Just the Main style sheet files
  - /img This contains modified images
  - /js Modified versions of all the default interface JavaScript files.  
Generally this involves ensuring that API calls use the root URL.  
Some additional changes have been made to pages using alpaca forms to omit the font stylesheets and where I have used buttons rather than the structures used by Mark. I have changed the name of some of these files to match their parent html file.
  - / lib contains the following full or partial libraries  
**datatables:** Fixed columns and a modified alt editor script.  
**HighCharts:** The 'ow-theme.js' file for charts  
**steelseries:** Contains the full library but only the 'gauges.js' file is required as this has had the tooltips modified to use the themes.
  - /themes contains the theme style sheets.

Any other files that the AI uses are accessed from the main interface folder so are unchanged.

Please read the ReadMe.txt file that you will find in the main AI2 folder. This details all changes.

## Appendix A

---

### *Adjusting fonts*

Responsive fonts appears to be a dark arts subject with many proposals. My solution is shown below:

```
body, p, label {  
    font-family: Inter, serif;  
    font-size: calc( 14px + 1 * (( 100vw - 300px) / ( var( --siteWidth ) - 300 )));  
}  
h1 { font-size: calc( 24px + 6 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }  
h2 { font-size: calc( 20px + 5 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }  
h3 { font-size: calc( 18px + 4 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }  
h4 { font-size: calc( 16px + 3 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }  
h5 { font-size: calc( 14px + 2 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }  
h6 { font-size: calc( 13px + 1 * (( 100vw - 300px) / ( var(--siteWidth) - 300 ))); }
```

The main values to edit are the first two - the base pixel size and the enlargement factor. Taking h1 as an example, the 24px represents the smallest size the font can be while 30px (24px + 6), is the largest it can be. The remaining calculation adjusts the enlargement factor based on the screen width. I believe the 300px represents the smaller screen width allowed and so the font will only be 24px high when the screen is 300px wide.

---

### *Page columns*

All pages use flex box children to display columns. These classes: 'ow-colOne' to 'ow-colSix' control the minimum and maximum width of the child 'div' elements. These are calculated based on the site width. However, as screen width reduces, some are modified to be 100% width to ensure the maximum screen width is used. Using the style 'flex: 1;' ensures that they expand to the maximum available.

As they are children of flex boxes, they can all have an 'order' style. This can be used to override the position of the child within the box. Children with the same order number are displayed in the sequence they are coded in the page, otherwise the order is as dictated by the order number.

---

## Charts

All the charts now use the theme selected for the site. This is supplied in the 'HighCharts' library area. This file can be edited but you should remember that it can only be used to change styles.

One feature of this script is that it defines the eight colours used by HighCharts on the 'select-a-chart-' pages. Therefore you can freely edit these colours as required. Colours used on the other charts are determined by CumulusMX. I have not investigated using the CumulusMX colours here but it should be possible.

---

## Editing Themes

All themes are easily edited. They start with a 'root' section that defines the colours used. This is shown below:

```
:root {
  --sub5: #f5f5f5;
  --sub4: #dddddd;
  --sub3: #bbbbbb;
  --sub2: #989898;
  --sub1: #767676;
  --theme: #555555;
  --add1: #4c4c4c;
  --add2: #434343;
  --add3: #3b3b3b;
  --add4: #323232;
  --add5: #2a2a2a;
  --modal: rgba(85, 85, 85, 0.3);
}
```

If you don't like a particular colour, then simply change its hex value and this change will be reflected for all styles that use the colour. (Note that the AI doesn't necessarily use all defined colours).

This is followed by a section that defines colour combinations for foreground and background. Again shown below:

```
/* Theme colour combinations (Foreground / Background) */
.ow-theme-add5 { color: var(--sub5); background-color: var(--add5); }
.ow-theme-add4 { color: var(--sub5); background-color: var(--add4); }
.ow-theme-add3 { color: var(--sub5); background-color: var(--add3); }
```



```
.ow-theme-add2 { color: var(--sub5); background-color: var(--add2); }  
.ow-theme-add1 { color: var(--sub5); background-color: var(--add1); }  
.ow-theme      { color: var(--sub5); background-color: var(--theme);}  
.ow-theme-sub1 { color: var(--add5); background-color: var(--sub1); }  
.ow-theme-sub2 { color: var(--add5); background-color: var(--sub2); }  
.ow-theme-sub3 { color: var(--add5); background-color: var(--sub3); }  
.ow-theme-sub4 { color: var(--add5); background-color: var(--sub4); }  
.ow-theme-sub5 { color: var(--add5); background-color: var(--sub5); }
```

If you find that a foreground / background combination doesn't work for you, simply change its variable ( `var(--xxxY)`), to one of the others. I have found that for some themes, I needed to use the lighter colour on the theme background while on others the dark colour was more suitable. Occasionally this needs to extend to backgrounds on either side of the main theme background.

It is also possible that you might want to modify the either combination colours for hovers.