SBSH PocketWeather Custom View Syntax

For PocketWeather 2.0.8

By: Adrian Oliver
Ronald A. Bernstein, M.D.

Last Updated: 13 June 2008

Downloads and additional support can be found at our site
http://www.sbsh.net

Overview

There are several ways to customize the display of data and images in PocketWeather. In addition to a host of Options, one can use a custom layout to create a custom view. The PocketWeather Today item (plugin) screen view and WeatherConsole both support views which can be customized, allowing you to configure PocketWeather to display weather data and images in pretty much any layout and style you wish.

When first installing PocketWeather, you are provided with a number of different layouts which can be selected from the MainMenu. These default layouts are organized into folders on your device, customized for either the Today screen view or WeatherConsole; however, there is nothing preventing you from selecting a Today screen view for WeatherConsole, or a WeatherConsole view for the Today screen -- they are interchangeable.

Obviously a Today screen view will not use the full display area available to WeatherConsole, while a WeatherConsole view may appear squashed or cropped into the limited Today screen display area and may require reformatting.

If you cannot find a layout that suits your needs from the default layouts or from the constantly growing supply of layouts that users are producing, then this document will help you to either tweak an existing layout or generate your own new custom view.

Files and Folders

PocketWeather custom views, or layouts, must end with the ".PWC" file extension. These are simple plain-text files which can be edited with your favourite text editor, such as Notepad. Although ANSI text encoding is allowed, it is strongly recommended that UNICODE (big endian) encoding be used.

To install a new custom view file onto your device, simply copy the ".PWC" file into the "Skins" folder on your device.

To make organising the custom view files easier, it is recommended that new folders (or sub-folders) be added under the "Skins" folder. For example, you could add a folder called "mine" under "skins".

The skin folders and files will appear in PocketWeather in terms of hierarchy and names as they are declared in the device's folder structure. This however also means that a folder called "Today Plugin" will always appear as "Today Plugin" in all locales. Same with file names.

To allow the translation of the folders and files, standard folders and files are named using numeric values which match onto PocketWeather's translation language strings. Below are the start IDs and their English string representations.

Folder String IDs

ID	English Language String
47000	"Today Extras"
47001	"Today Plugin"
47002	"WeatherConsole"
47003	"Non Tabbed"

File String IDs

ID	English Language String
47100	"Detailed"
47101	"Summary"
47102	"Classic"
47103	"Classic Forecast"
47104	"Classic Navigator"
47105	"Compact"
47106	"Compact Forecast"
47107	"Single Line"
47108	"MultiLine 2"
47109	"MultiLine 3"
47110	"MultiLine 4"
47111	"Bold Horizontal"
47112	"Bold Horizontal Large Icons"
47113	"Bold Horizontal Small Icons"
47114	"Bottom Horizontal"

ID	English Language String
47115	"Classic with Analogue Clock"
47116	"Classic with Digital Clock"
47117	"Classic No Images"
47118	"Classic No Images with Units"
47119	"Classic with Units"
47120	"Compact Retro"
47121	"Compact Vertical"
47122	"Compact Vertical with Name"
47123	"DOS"
47124	"Retro"
47125	"Vertical"
47126	"Vertical Retro"
47127	"Classic No Images" (see note below)
47128	"Classic No Images with Units" (see note below)
47129	"Classic" (see note below)
47130	"Classic with Units" (see note below)
47131	"Compact" (see note below)
47132	"Compact Retro" (see note below)
47133	"Moon"
47134	"Alerts"
47135	"METAR"

Note: some strings may appear duplicates in the above table. This IS deliberate to overcome the CAB file installation issue where each filename has to be unique, even when in different folders.

If you would like additional translation string IDs to be added to PocketWeather's translation table, please send any suggestions to adrian.oliver@sbsh.net.

File Content

The custom layout text file contain a series of instructions or commands that tell PocketWeather where to place the various elements that make up the display.

Each command must begin with a semicolon (";"), followed immediately with the command name in capital letters. Lower case or mixed case command are ignored and may produce unexpected results.

Immediately following the command name on the next line are the various command specific attributes, for example, position, font, color, etc. These attributes will typically span several consecutive lines; there must not be any empty lines between these attributes.

The command and its attributes combine to form an object.

A layout script will consist of multiple objects. Each object must be separated by at least one blank/empty line.

By adding objects of different types and different attributes into a script, one can construct a custom layout.

Custom Layout Script - A Simple Example

Below is a simple example of a custom layout script which contains 6 objects which will be described later in this document.

```
;HEIGHT
45

;IMAGE
#ICON
10,10,20,20,c

;TEXT
#TEMP#
50,00,20,15,c

;TEXT
#HIGH#
50,10,20,15,c

;TEXT
#LOW#
50,20,20,15,c

;TEXT
Welcome to PocketWeather Scripting
50,30,190,15,c
```

This produces the following output

Objects

The multiline entity consisting of the *;COMMAND (this declares the object type)*, its position, and other necessary attributes or elements is termed an OBJECT.

There are various objects that PocketWeather scripts commonly employ for displaying text, graphics, images, drawing shapes, lines and shaded areas, graphs, displaying variables and strings, and navigating to pages.

All objects must be preceded by an empty line, and must be followed by an empty line.

If one wishes to prevent a particular command from executing, it can be disabled by converting its object into a **comment**. See below.

Comments

A comment is defined as having either a "-" before and/or after the first line of the object.

For example

> *- This is a comment*
>
> *-This is also a comment*
>
> *So is this a comment-*
>
> *;DRAW-*
> *RECTANGLE PENCOLOR-0-0-0 BRUSHCOLOR-250-250-240 2*
> *0%,14,100%,64*

The above object is completely commented out because the ";DRAW" has a "-" afterwards, resulting in the complete object being disabled (i.e. the next two lines).

For software developers, "//" is also a comment – e.g. "// this is a comment"

Indentation

Leading white space and tabs are ignored.   This means that the scripting can be suitably indented to help code readability.

Object Positioning and Dimensions

X,Y,W,H,<Horizontal Alignment>

Dimensions can be absolute values, "W" or "H" for width or height, or percentage of dimension - e.g. 50% means 50% of width or height.

Example,

0, 0, 100%, 14
(starts at 0,0, 100% of width, 14 pixels height)

or

2%, 2%, 98%, 60%
(starts at 2% of width, 2% of height, and takes up remains of the width, and 60% of height)

or

10, 10, W-20,H-20
(starts at 10, 10, and takes up all the width and height, except for last 10 pixels horizontally and vertically).

or

2%+2,10%+100, 16, 16
(starts at 2% of width plus 2 pixels, 10% of height plus 100 pixel).

Syntax	Description
C	for Center (horizontal) aligned
L	for Left aligned
R	for Right aligned
W	for word Break - cause text to go across multiple lines as appropriate
T	for Top aligned
B	for Botton aligned
V	for Vertical centered aligned
E	truncates any word that does not fit in the rectangle and adds ellipses – WM5 and above only (uses the Microsoft DT_WORD_ELLIPSIS flag)

Note that Bottom and Vertical Centered CANNOT be used for word break.

Word break only applies to text objects.

Applies to text and graphics objects.

Parameters can be combined – for example, "CV", or "TR"

Colors

Color are specified using the standard R-G-B format, where:

- R is the red component (0-255)
- G is the green component (0-255)
- B is the blue component (0-255)

where "0-0-0" produces black, and "255-255-255" produces white.

When producing custom views for the today screen, specifying "TODAY" will cause it to use the device's configured today screen text color.

NOTE: Colors MUST be specified as "R-G-B" - "R,G,B" is not accepted, and may cause unexpected results.

Font Definition

> <Font Name>,<Size>,<Font weight>,<Color>,<Rotation>

All fonts are defined using the above syntax.

Attribute	Comment
Font Name	e.g. "Tahoma". Can be any name of fonts installed on the device. In addition, any TTF font file located in the same folder as the layout file can also be referenced.
Size	e.g. "12". The font point size
Font Weight	Valid attributes: ● "NORMAL" ● "BOLD" ● "UPPER" ● "LOWER" ● "SHADOW" ● "GLOW-r-g-b" ● "GLOWEXPAND-n" ● "ITALIC" ● "UNDERLINE"
Color	e.g. "12-34-56", or "TODAY" to use the device's today screen font color.
Rotation	Optional text rotation in degrees. Default is 0.

Examples:

Tahoma,10,NORMAL,TODAY

Tahoma,10,NORMAL,TODAY,90

Bradley Hand ITC,13,UPPERCASE GLOW-100-51-100 GLOWEXPAND-2 BOLD,255-255-255

Heading

Syntax	Description
;TITLE <some name>	For WeatherConsole, this specifies the name of the tab Ignored for Today Screen view
;ICONID 122	Should always be this for WeatherConsole. Ignored for Today Screen view
;ICON fred.tga	Specify custom WeatherConsole tab icon. Can be TGA or BMP file. Should be located in the current script folder.
;HEIGHT 123	The page height Must specify this for Today Screen. For WeatherConsole, if this is specified, then the user can scroll/pane vertically
;WIDTH 240	The page width No need to specify this for Today Screen. For WeatherConsole, if this is specified, then the user can scroll/pane horizontally. If you specify a width, then the view becomes fixed width and does not automatically resize
;SETTINGS <icon image> 0,0	Draws the settings menu button at the specified position
;DRAWTABS <Font Name>,<Size>,<Font weight>,<Color> <Tab Shading % - optional> <Tab Color – optional> <Icon Padding – optional> <Location Name Position> <"ANALOG"/"DIGITIAL" Tab Clock – optional>,<Digital Tab Clock Position> <Digital Tab Clock Font Name>,<Size>,<Font weight>,<color>	See "DRAWTABS" section below
;NODRAWTABS	See "DRAWTABS" section below
;NODRAWTOPHEADINGBAR	WeatherConsole Only: disables the displaying of the top heading bar which displays the location name and date/time
;DEBUG	This will display object rectangles in black and the touch areas in red. This allows the developer to see the positioning and sizing of each object.
;DEBUGOBJECTS	This will display object rectangles in black. Touch areas are not drawn
;DEBUGTOUCH	This will display object touch areas in red. Object rectangles are not drawn.
;DEBUGEXECUTION	This will output into the system log file the object execution path
;RELOAD <icon image> x,y	Causes the custom layout script to be reloaded
;REFRESH <icon image> x,y	Causes weather and image update process to start
;REFRESH-WEATHER <icon image> x,y	Causes weather update process to start

Syntax	Description
;REFRESH-IMAGE <icon image> x,y	Causes image update process to start
;UP <icon image> x,y	Location change – select previous location, display icon
;DOWN <icon image> x,y	Location change – select next location, display icon
;HIDPI	Flags that the layout script dimensions were designed for VGA/HiDPI devices. When running on a QVGA device, dimensions should be scaled appropriately.
;ROTATE-START <icon image> x,y	Today Screen Only: Starts location rotation.
;ROTATE-STOP <icon image> x,y	Today Screen Only: Stops location rotation.
;ROTATE <icon image> x,y	Today Screen Only: Toggles location rotation
;ROTATE-START-FORCE	Today Screen Only: Starts location rotation immediately on the initial run of the script
;ROTATE-STOP-FORCE	Today Screen Only: Stops location rotation immediately on the initial run of the script

DRAWTABS

Location Tabs - Today Screen Only (PPC/Classic/Professional devices only – i.e. Not SmartPhones/Standard)

The control of the today screen location tabs is described in summary in the previous Heading section. This section provides additional details.

Without any layout command, Location Tabs will appear with default values controlled only by the configured options in PocketWeather Options under "Today".

(Default – no clock or weather icons)   (Default – analog clock and weather icons)

To explicitly disable tabs from appear, use the "**;NODRAWTABS**" command in the header.

Syntax:

> *;NODRAWTABS*

To explicitly enable tabs to appear, use the "**;DRAWTABS**" command in the header.

Syntax:

> *;DRAWTABS*
> *<Font Name>,<Size>,<Font weight>,<Color>*
> *<Tab Shading %>*
> *<Tab Color >*
> *<Icon Padding>*
> *<Location Name Position>*
> *<"ANALOG"/"DIGITIAL" Tab Clock>,<Digital Tab Clock Position>*
> *<Digital Tab Clock Font Name>,<Size>,<Font weight>,<Color>*

Syntax	Description
<Font Name>,<Size>,<Font weight>,<Color>	see above section on fonts
<Tab Shading %>	As percent. If tab shading is >0, then unselected location tabs are shared. If tab shading is <0, then selected location tab is shared
<Tab Color>	Tab color – if not specified, then uses default color.

Syntax	Description
<Y Padding> OR <X, Y Padding>	The tab Padding is affects the total tab width and height – and is the additional width and height added to the minimum tab width and height required by the settings button icon, font height and width, and analogue/digital clock. Note that if only one value is specified (I.e. "20"), then for legacy reasons, this is taken to be the vertical padding. * See Padding Note below
<LocationNamePosition>	Specifies the position of the location name within the tab. See above section on "Object Positioning and Dimensions". Valid position specifiers, "T,V,B,L,C,R"
<TabClock>,<DigitalTabClockPosition>	"**ANALOG**" - displays analogue clock to the left of the location's name. The Analogue clock will ALWAYS be displayed on the far left. "**DIGITAL**" - displays a digital clock. Default position is below and to the right of the location's name. See above section on "Object Positioning and Dimensions". Valid position specifiers, "T,V,B,L,C,R" "**NOCLOCK**" - disables the display of clock.
<DigitalTabClockFontName>,<DigitalTabClockFontSize>, <DigitalTabClockFontweight>,<DigitalTabClockFontColor>	see above section on font
<TabWeatherIcon>,<TabWeatherIconAlignment>	"**FORECASTICON**" - displays the current day's weather icon "**CURRENTICON**" - displays the current condition's weather icon Can also specify position. See above section on "Object Positioning and Dimensions". Valid position specifiers, "T,V,B,L,C,R" **Note** that if the height of the weather icons is greater than the height of the tab, the weather icon will not be displayed. The currently configured "Mini" icons will always be used. If the weather icon fails to appear, increase the Y padding.

Padding Note – PocketWeather is unable to guess the desired layout being designed, and therefore will only ensure minimum horizontal space to display the location's name and the optional clock (digital or analogue). Therefore the script writer MUST ensure enough vertical and horizontal space using the X-Y padding to allow space for the location name, option clock, and weather icon. If undesirable overlapping occurs, increase the X and/or Y padding.

Example:

> *;DRAWTABS*
> *Tahoma,10,BOLD,TODAY*
> *28*
> *TODAY*
> *16,0*
> *RT*
> *DIGITAL,RB*
> *Tahoma,10,N,TODAY*
> *CURRENTICON,LV*

Example:

> *;DRAWTABS*
> *Courier New,12,BOLD ITALIC UNDERLINE,TODAY*
> *28*
> *TODAY*
> *0*
> *RV*
> *ANALOGUE*

Image Objects

Syntax:

> *;IMAGE*
> *<Image object type>*
> *X, Y, W, H, <Horizontal alignment>*

Where type can be:

"<Filename>" - must be BMP or TGA file
"<Filename-QVGA>,<Filename-VGA>" - will auto select the QVGA/VGA BMP/TGA file depending on device's current DPI setting
"#ICON" or "#DAY-ICON" - the weather forecast icon for the current forecast day - adding "WIND" causes wind direction arrow to be displayed. Adding "NOWIND" disables/stops wind direction arrows from being displayed.
"#RADAR-ALL" - displays all the downloaded images enabled for that location
"#RADAR-n" - displays the n'th downloaded image for that location - zero based.
"#CURRENTICON" - the current condition weather icon - adding "WIND" causes wind direction arrow to be displayed
"#MOON" - the current day's moon phase icon
"#WIND" - displays the wind compass and wind direction arrow
"#WINDARROW" - displays just the wind direction arrow
"#300#" - displays image from the stock image resources

Icon Sizes

With the following command:

> *;IMAGE*
> *#ICON*
> *0,0*

it will automatically select the default configured icon.

If you want to explicitly control the icon size, this can be achieved using the following specifiers after the "#ICON" keyword to select the different configured icon sizes

- MINI   - 16x16 for QVGA, 32x32 for VGA
- NORMAL   - 32x32 for QVGA, 64x64 for VGA
- LARGE   - 64x64 for QVGA, 128x128 for VGA
- GIANT   - 96x96 for QVGA, 192x192 for VGA

For example
> *;IMAGE*
> *#ICON-MINI*
> *0,0*
>
> *;IMAGE*
> *#ICON-LARGE*
> *30,0*

Day/Night Icon

With the following command:

;IMAGE
#ICON
0,0

it will automatically select the correct day/night icons depending on the location's local time.

By default, all forecast weather icons use day-time icons.

You can forecast the use of day or night icons by added "-DAY" or "-NIGHT" to the icon command.

For example

;IMAGE
#ICON-DAY
0,0

;IMAGE
#ICON-MINI-NIGHT
0,0
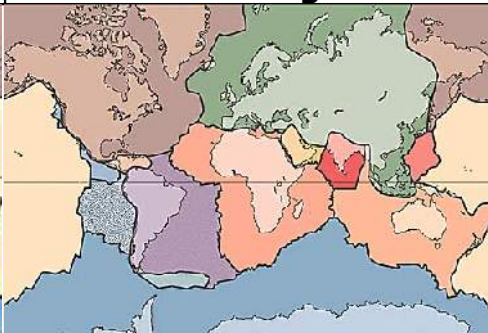
;IMAGE
#ICON-LARGE-DAY
30,0

Stock Image IDS

ID	Description	QVGA Images	VGA Images
101	Pocket Weather Logo non-transparent		
102	Pressure Icon		
104	Visibility Icon		
105	Dewpoint Icon		
106	Precipitation Icon		
107	Thermometer Icon		
108	Map Zoom In		
109	Map Zoom Out		

ID	Description	QVGA Images	VGA Images
110	Map Rotate		
111	Map animation controller – play		
112	Map animation controller – pause		
113	Left arrow		
114	Right arrow		
115	Selected Icon		
116	WeatherConsole Title Icon		
117	Wind Arrow 1 (small)	(wind_arrow_1.png)	(wind_arrow_vga_1.png)
118	Wind Arrow 2 (medium)	(wind_arrow_2.png)	(wind_arrow_vga_2.png)
119	Wind Arrow 3 (large)	(wind_arrow_3.png)	(wind_arrow_vga_3.png)
120	Compass	(compass.png)	(compass_vga.png)
121	WeatherConsole Earthquake Tab Icon	(earthquakes_tab.png)	(earthquakes_tab_vga.png)
122	WeatherConsole Summary Tab Icon	(Summary_tab.png)	(Summary_tab_vga.png)
123	WeatherConsole METAR Tab Icon	(METAR_tab.png)	(METAR_tab_vga_png)
124	WeatherConsole Forecast Tab Icon	(Forecast_tab.png)	(Forecast_tab_vga.png)
125	WeatherConsole Forecast Compare Tab Icon	(Forecasts_tab.png)	(Forecasts_tab_vga.png)
126	WeatherConsole Maps Tab Icon	(maps_tab.png)	(maps_tab_vga.png)

ID	Description	QVGA Images	VGA Images
127	WeatherConsole Maps Compare Tab Icon	(mapscom_tab.png)	(mapscom_tab_vga.png)
130	Folder Selection Dialog – Up Icon		
131	Folder Selection Dialog – Toolbar Icon		
132	WeatherConsole Home Icon	(home.png)	(home_vga_png)
133	Current Temperature Icon		
140	Down Arrow Icon		
141	Up Arrow Icon		
150	Settings Button – Up-To-Date	(updated.png)	(Updated_vga.png)
152	Settings Button – Out-Of-Date	(Outdated.png)	(Outdated_vga.png)
153	Settings Button – Updating	(Updating.png)	(Updating_vga.png)
160	Alarm Bell		
161	Alert Icon		
200	Earthquake Map	(worldmap.jpg)	(worldmap_vga.jpg)

ID	Description	QVGA Images	VGA Images
201	Earthquake Tectonic Map	(worldmap2.jpg)	(worldmap2_vga.jpg)
300	Sun Rise/Set Background image		
400	PocketWeather Config main image list		

** NOTE - depending on the device's DPI settings, using these will automatically select the VGA or QVGA image size

Draw Objects

Syntax:

> ;DRAW
> RECTANGLE PENCOLOR-<PENCOLOR> BRUSHCOLOR-<BRUSHCOLOR> WIDTH-<Line Width>
> ALPHA-<Shading>
> X, Y, W, H

or

> ;DRAW
> RECTANGLE PENCOLOR-<PENCOLOR> BRUSHCOLOR-<BRUSHCOLOR> WIDTH-<Line Width>
> X, Y, W, H

or

> ;DRAW
> LINE PENCOLOR-<PENCOLOR> WIDTH-<Line Width>
> X, Y, X-Direction, Y-Direction

Example:

> ;DRAW
> RECTANGLE PENCOLOR-0-0-0 BRUSHCOLOR-250-250-240 WIDTH-2
> 0%,14,100%,64

or

> ;DRAW
> LINE PENCOLOR-0-0-0 WIDTH-2
> 40%, 14, 0, 64

> ;DRAW
> RECTANGLE PENCOLOR-0-0-0 BRUSHCOLOR-250-250-240 WIDTH-2 ALPHA-50
> 0%,14,100%,64

This draws a rectangle 50% shaded/blended with the background.

Note that for line objects, the X/Y directions can be positive or negative.

Draw Objects With Variable Size

The simple draw objects described in the previous section can have their dimensions modified at runtime based on weather variables such as current temperature, wind speed, pressure, etc.

The syntax for these draw objects with variable size is identical to the normal draw objects, with the additional of extra attributes.   These attributes can apply to both lines and rectangles.
In addition to the rectangle and line color and width attributes, the following options attributes can be user to change the height, width or length of the objects at run time based on weather variables such as current temperature, wind speed, etc.

The syntax of the additional attributes is:

Name	Description
VARIABLE-<constant/variable>	Declares the value or constant to use. For example, current temperature.
VARIABLEMAX-<constant/variable>	Declares the maximum range of the variable. This can be a fixed constant value, or another value such as temperature high limit.
VARIABLEMIN-<constant/variable>	Declares the minimum range of the variable. This can be a fixed constant value, or another value such as temperature low limit.
VARIABLEDIRECTION-<direction>	***Rectangle draw objects only***. Defines which direction to scale. Valid entries are "UP", "DOWN", "LEFT" and "RIGHT. For example, "VARIABLEDIRECTION-UPDOWNLEFTRIGHT", or " VARIABLEDIRECTION-UP", or " VARIABLEDIRECTION-RIGHT". The default is "UP". For line objects, the default operation is to always start at the defined starting point, and to scale to length from that point in the specified direction.

Example:

> *;DRAW*
> *LINE PENCOLOR-121-163-187 WIDTH-10 VARIABLE-TEMP VARIABLEMIN-LOW VARIABLEMAX-*
> *HIGH*
> *0,14,100%,0*

This would result in a line starting on the left-hand size and extending horizontally to the right, where the length would be based on the current temperature, between the limits of the high and low temperatures. So if the current temperature was 20'C, the minimum temperature was 10'C, and the maximum temperature was 30'C, the line would start on the left, and extend 50% (I.e. Half of 100%) across to the right.

Example:

> *;DRAW*
> *RECTANGLE PENCOLOR-121-163-187 WIDTH-10 VARIABLE-TEMP VARIABLEMIN-LOW*
> *VARIABLEMAX-HIGH VARIABLEDIRECTION-RIGHT*
> *0,0,100%,100%*

This would result in a rectangle starting in the top left corner, extending from the top to the bottom fully, but only extending to the right by the amount defined by the current temperature within the range of the forecast high and low temperatures.

Supported Variables

Name	Description	Associated Weather Template Variable
CITYID	The location's index ID	
BATTERYTEMP	the device's current battery temperature - availability depends on device's hardware	
BATTERYLIFTPERCENTAGE	the device's current battery life percentage - availability depends on device's hardware	
TEMP	temperature (current, or forecast, as appropriate)	"tmpr" and "tmprUnits"
FEELSLIKETEMP	effective temperature	(calculated)
HUMIDITY	Relative Humidity	"humidity"
DEWPOINT	Dew point	"dewPoint" and "tmprUnits"
RAIN	Rain Probability % (I.e. change of rain falling in the future)	"precipitation"
RAINFALLAMOUNT	Amount of rain recorded (I.e. what has already fallen)	"rainfall" and "rainfallUnits"
BAROMETER	the barometer value	"pressure" and "pressureUnits"
VISIBILITY	Visibility distance	"visibility" and "visibilityUnits"
WINDDIR-NUMBER	wind direction in degrees - e.g. 0' to 360'	"windDirection"
WINDSPEED	Wind speed	"windSpeed" and "windSpeedUnits"
WINDGUSTSPEED	Wind gust speed	"windGustSpeed" and "windSpeedUnits"
HIGHAPPARENT	apparent forecast high/max/day temperature	(calculated)
LOWAPPARENT	apparent forecast low/min/night temperature	(calculated)
HIGH	forecast high/max/day temperature	"tmprHi" and "tmprUnits"
LOW	forecast low/min/night temperature	"tmprLo" and "tmprUnits"
MOON-PERCENT	moon phase %	(calculated)

Note that for variables with associated units, the following default units are assumed:

Temperature	Degrees C
Pressure	mBar
Speed	KPH
Visibility Distance	Kilometres
Elevation	Meters
Depth	Kilometres

Therefore, when when specifying a variable with fixed limits, you should assume the about units and use the appropriate units.

For example, for temperature bar:

;DRAW
RECTANGLE PENCOLOR-121-163-187 WIDTH-10 VARIABLE-TEMP VARIABLEMIN--10
VARIABLEMAX-40 VARIABLEDIRECTION-RIGHT
0,0,100%,100%

Graph Object

Syntax:

> ;GRAPH
> <COMMANDS>
> X, Y, W, H

The "<COMMANDS>" can contain any of the following commands:

Type Command (only one can be specified)	Description
TEMP	Displays temperature graph
RAIN	Displays rain trending (subject to availability from weather feed)
HUMIDITY	Displays humidity trending (subject to availability from weather feed)
BAROMETER	Displays barometric pressure trending (subject to availability from weather feed)
ICON	Displays weather icons for each day and any significant weather change.
LEGEND	Just draws the graph background legend.

Only one of the above commands can be specified.

Each of these commands can be have the following display attributes – note that conflicting attributes are ignored (I.e. DAILY and HOURLY...)

Display Attributes	Description
DAILY	Disables the hourly forecast
HOURLY	Enables the hourly forecast
"N"HOUR	Displays the next N hours – e.g. "24hours". Note that there MUST be no gap between the number of hours and the word "HOURS"
"N"DAYS	Displays the next N days - .e.g. "5days". Note that there MUST be no gap between the number of days and the word "DAYS"
NOLEGEND	Disables graph background legend display
COLOR-R-G-B	R-G-B color of the graph line.

Example:

> ;GRAPH
> TEMP 48HOURS COLOR-0-255-0
> 0,0,100%,100%
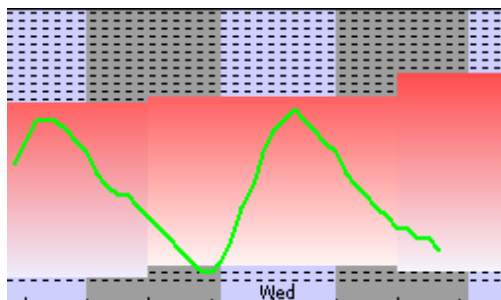
Using the "LEGEND" command, you can build up map overlays.

Example:

> *;GRAPH*
> *48HOURS LEGEND*
> *0,0,100%,100%*
>
> *;GRAPH*
> *TEMP 48HOURS COLOR-0-255-0 NOLEGEND*
> *0,0,100%,100%*
>
> *;GRAPH*
> *HUMID 48HOURS COLOR-0-255-255 NOLEGEND*
> *0,0,100%,100%*
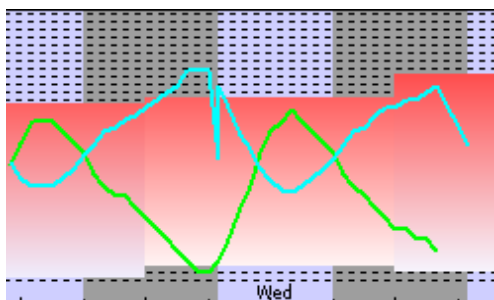
Text Object

Syntax:

> *;TEXT*
> *<The String>*
> *X,Y,W,H,<Alignment>*
> *<Font Name>,<Size>,<Font weight>,<Color>*

<The String> can be any free format string.

The following tables list the variable reserved.

Date/Time Objects

Note:
1. ALL times and dates are assumed to be in device/user local time unless explicitly specified.
2. ALL components refer to the currently active location unless specified or not appropriate.
3. ALL time and date components refer to the currently active location, and to the currently selected weather item (i.e. current conditions, or forecast data).
4. "#DATE" objects, when applied to daily forecasts, will not have any UTC to user/device/location local time conversion applied, because they always refer to the forecast date. In such cases, "#DATE-DAYOFWEEK-LOCAL#" and "#DATE-DAYOFWEEK#" will always display the same string regardless of the timezone the user/device is in and the location's timezone.

Name	Description
#UPDATE-TIME#	the most recent update time, whatever was most recently updated (i.e. current conditions, forecast, images)
#UPDATE-TIME-FORECAST#	the last forecast update time
#UPDATE-TIME-CURRENT#	the last current conditions update time
#UPDATE-TIME-IMAGES#	the last images update time
#UPDATE-TIME-EARTHQUAKE#	The last earthquake update time
#UPDATE-SCHEDULE-NEXT#	The next scheduled update – scheduler must be enabled
#UPDATE-TIME-NEXT#	The next update time. If the scheduler is enabled, then displays the next earliest scheduled update time, otherwise the next update time based on the existing updates and the appropriate update rates.
#TIME#	The current time
#TIME-LOCAL#	The current time in location local time
#TIME-METAR#	The time when the METAR observation data was recorded
#TIME-METAR-LOCAL#	The time when the METAR observation data was recorded in location local time.
#TIME-FORECAST#	The current forecast time. Only meaningful for hourly forecast data.
#TIME-FORECAST-LOCAL#	The current forecast time in location local time. Only meaningful for hourly forecast data. (see note 4 above)
#DATE-DAYOFWEEK#	Day of week number – e.g. "1" (Monday = 1, etc)
#DATE-DAYOFWEEK-LOCAL#	Day of week number in location local time – e.g. "1" (Monday = 1, etc) (see note 4 above)

Name	Description
#DATE-DAYOFWEEK-STRING#	Brief version of the day string – e.g. "Mon"
#DATE-DAYOFWEEK-STRING-LOCAL#	Brief version of the day string in location local time – e.g. "Mon" (see note 4 above)
#DATE-DAYOFWEEK-STRING-LONG#	Long version of the day string – e.g. "Monday"
#DATE-DAYOFWEEK-STRING-LONG-LOCAL#	Long version of the day string in location local time – e.g. "Monday" (see note 4 above)
#DATE-DAY#	Day of month
#DATE-DAY-LOCAL#	Day of month in location local time (see note 4 above)
#DATE-DAY-LONG#	Day of month with leading "0" if appropriate
#DATE-DAY-LONG-LOCAL#	Day of month in location local time with leading "0" if appropriate (see note 4 above)
#DATE-MONTH#	Month number
#DATE-MONTH-LOCAL#	Month number in location local time (see note 4 above)
#DATE-MONTH-LONG#	Month number with leading "0" if appropriate
#DATE-MONTH-LONG-LOCAL#	Month number in location local time with leading "0" if appropriate (see note 4 above)
#DATE-MONTH-STRING#	Short Month string – e.g. "Jan"
#DATE-MONTH-STRING-LOCAL#	Short Month string in location local time – e.g. "Jan" (see note 4 above)
#DATE-MONTH-STRING-LONG#	Month string – e.g. "January"
#DATE-MONTH-STRING-LONG-LOCAL#	Month string in location local time – e.g. "January" (see note 4 above)
#DATE-YEAR-LONG#	Year number– e.g. "2010"
#DATE-YEAR-LONG-LOCAL#	Year number in location local time– e.g. "2010" (see note 4 above)
#DATE-YEAR#	Year number– e.g. "08"
#DATE-YEAR-LOCAL#	Year number in location local time– e.g. "08" (see note 4 above)
#DATE-LONG#	The long version of the date string as specified by the device's regional settings and any user customization
#DATE-LONG-LOCAL#	The long version of the date string in location local time as specified by the device's regional settings and any user customization (see note 4 above)
#DATE#	The short version of the date string as specified by the device's regional settings and any user cujstomization
#DATE-LOCAL#	The short version of the date string in location local time as specified by the device's regional settings and any user customization (see note 4 above)

General Variables

Note that the following variables are either result from the specific location's configuration, or derived/calculated.

Name	Description
#METARSTATIONNAME#	the currently selected METAR station name, or ICAO code if the name is not available
#METARSTATIONICAO#	the currently selected METAR station ICAO code
#METAR#	Outputs the full raw METAR data for the currently selected ICAO station
#METAR-n#	Outputs the full raw METAR data for ICAO station "n"
#TAF#	Outputs the full raw TAF data (short term forecast) for the currently selected ICAO station
#TAF-n#	Outputs the full raw TAF data (short term forecast) for ICAO station "n"
#CITY#	Displays the location name
#CITYID#	Displays the location's ID – this is the location's position within the user's list of locations.
#CITYCODE#	Displays the location's country code
#CITYCOUNTRY#	Displays the location's country name
#ASTRONOMICALTWILIGHTSTART#	Twilight Start (before sunrise) - sun is 18' below the horizon
#NAUTICALTWILIGHTSTART#	Twilight Start (before sunrise) - sun is 12' below the horizon
#CIVILTWILIGHTSTART#	Twilight Start (before sunrise) - sun is 6' below the horizon
#TWILIGHTSTART#	Civil Twilight Start (before sunrise) - same as CIVILTWILIGHTSTART
#SUNRISE#	sun rise
#SUNNOON#	Noon – when the sun is at its highest
#SUNSET#	sun set
#TWILIGHTEND#	Twilight End (after sunset) - same as CIVILTWILIGHTEND
#CIVILTWILIGHTEND#	Twilight End (after sunset) - sun is 6' below the horizon
#NAUTICALTWILIGHTEND#	Twilight End (after sunset) - sun is 12' below the horizon
#ASTRONOMICALTWILIGHTEND#	Twilight End (after sunset) - sun is 18' below the horizon
#SUNHOURS#	number of daylight hours
#MOON-PERCENT#	moon phase %
#MOON-TEXT#	moon phase text
#S-DEG#	the temperature units string
#S-SPEED#	the speed units string
#S-DIST#	the distance units string
#nnnnn#	extracts the appropriate language resource string from PocketWeather's language resource strings - definition of IDs is below
#BATTERYTEMP#	the device's current battery temperature - availability depends on device's hardware
#BATTERYLIFTPERCENTAGE#	the device's current battery life percentage - availability depends on device's hardware

SBSH

Current Conditions Weather Template Related Variables

Note that despite the following variable being available, not all current conditions weather feeds provide all entries. Please check with the particular current conditions weather feeds as to which of the following provide meaningful data.

Name	Description	Associated Weather Template Variable
#TEXT#	the current weather description text (i.e. current conditions, or appropriate day's forecast)	Derived from "sky"
#TEMP#	temperature (current, or forecast, as appropriate)	"tmpr"
#FEELSLIKETEMP#	Effective temperature	(calculated)
#HUMIDITY#	Relative Humidity	"humidity"
#DEWPOINT#	Dew point	"dewPoint"
#RAINFALLAMOUNT#	The amount of rain fall recorded – includes units	"rainfall" and "rainfallUnits"
#BAROMETER-TEXT#	"Rising"/"Steady"/"Falling"	(calculated)
#BAROMETER#	the barometer value	"pressure" and "pressureUnits"
#VISIBILITY#	Visibility distance	"visibility" and "visibilityUnits"
#WINDDIR-TEXT#	wind direction text	"windDirection"
#WINDDIR-NUMBER#	wind direction in degrees - e.g. 0' to 360'	"windDirection"
#WINDSPEED#	Wind speed	"windSpeed"
#WINDSPEEDUNITS#	Wind speed, plus appends the appropriate units. If no wind, displays "Calm"	"windSpeedUnits"
#WINDGUSTSPEED#	Wind gust speed	"windGustSpeed"
#WINDGUSTSPEEDUNITS#	Wind gust speed, plus appends the appropriate units.	"windSpeedUnits"

Forecast Weather Template Related Variables

Note that despite the following variable being available, not all forecast weather feeds provide all entries. Please check with the particular forecast weather feeds as to which of the following provide meaningful data.

Name	Description	Associated Weather Template Variable
#TEXT#	the current weather description text (i.e. current conditions, or appropriate day's forecast)	Derived from "sky"
#HUMIDITY#	Relative Humidity	"humidity"
#DEWPOINT#	Dew point	"dewPoint"
#RAIN#	Rain Probability	"precipitation"
#WINDDIR-TEXT#	wind direction text	"windDirection"
#WINDDIR-NUMBER#	wind direction in degrees - e.g. 0' to 360'	"windDirection"
#WINDSPEED#	Wind speed	"windSpeed"
#WINDSPEEDUNITS#	Wind speed, plus appends the appropriate units. If no wind, displays "Calm"	"windSpeedUnits"
#WINDGUSTSPEED#	Wind gust speed	"windGustSpeed"
#WINDGUSTSPEEDUNITS#	Wind gust speed, plus appends the appropriate units.	"windSpeedUnits"
#HIGH#	forecast high/max/day temperature	"tmprHi"
#LOW#	forecast low/min/night temperature	"tmprLo"
#HIGHLOW#	forecast max/min formatted appropriate	"tmprHi" and "tmprLo"
#HIGHAPPARENT#	apparent forecast high/max/day temperature	(calculated)
#LOWAPPARENT#	apparent forecast low/min/night temperature	(calculated)

Weather Alert Template Related Variables

Note that despite the following variable being available, not all weather feeds provide all entries. Please check with the particular weather feeds as to which of the following provide meaningful data.

Name	Description	Associated Weather Template Variable
#ALERT-FORECAST#	Displays forecast alert message – depends on feed and configuration. This will be generated by the METAR feed	Extracted from METAR feed
#ALERT-CURRENT-CONDITIONS#	Displays any and all severe current condition weather alerts for all the location's METAR stations - depends on alert configuration.	Extracted from METAR feed
#ALERT-CURRENT-CONDITIONS-n#	Displays any severe current condition weather for a particular METAR station, where n is the zero based station index.	Extracted from METAR feed
#ALERT-n#	Displays the content of the "alertn" variable provided by the weather feed template, where "n" is "0", "1"...	"alertTextn"
#ALERT#	Displays the content of all alert strings available (forecast, current conditions, or "alertn" variables, depending on the weather feed.	"alertText"
#USER-n#	Displays the content of the "user-n" variable provided by the weather feed template, where "n" is "0", "1"...	"userTextn"

Earthquake Variables

The following variables are extracted from the fixed earthquake feed.

Name	Description
#EARTHQUAKE-DATE#	displays date of latest earthquake
#EARTHQUAKE-MAG#	displays magitude of latest earthquake
#EARTHQUAKE-LOC#	displays location of latest earthquake
#EARTHQUAKE-NEAR#	displays nearest town/city to latest earthquake
#EARTHQUAKE-n-DATE#	displays date of latest-n earthquake
#EARTHQUAKE-n-MAG#	displays magitude of latest-n earthquake
#EARTHQUAKE-n-LOC#	displays location of latest-n earthquake
#EARTHQUAKE-n-NEAR#	displays nearest town/city to latest-n earthquake

Example:

> *;TEXT*
> *#TEMP#°#S-DEG#*
> *0,14,20%,10,C*
>
> *;TEXT*
> *#30010#*
> *80%,15,20%,12,c*

String IDS

These are automatically translated into the configured language.

Example:

> *;TEXT*
> *#30015#*
> *0,14,20%,10,C*

30001	"Sun"
30002	"Moon"
30004	"Precipitation"
30005	"Sunrise"
30006	"Zenith"
30007	"Sunset"
30008	"Actual"
30009	"Apparent"
30010	"Wind"
30011	"Visibility"
30012	"Humidity"
30014	"Current Conditions"
30015	"Today's forecast"
30016	"Day"
30017	"Night"
30019	"Images"
30027	"RH"
30029	"Pptn"
30031	"Pressure"
30032	"Maximum"
30033	"Minimum"
30018	"Recent Earthquakes"
30133	"Downloaded at"
31680	"Next"
31681	"Previous"
31682	"Start"
31683	"Stop"
30077	"at"
31667	"Hourly Forecast"
30043	"downloaded"
30078	"Gust"
30079	"Gusting"
30085	"No active alerts available"
30325	"Close"
30338	"Dismiss"
30309	"Exit"
30308	"Apply"
30100	"Max"
30101	"Min"
30103	"Reset"
30042	"observed at"

Location Command Object

These objects are used to modify the current working set within the custom layout.

;LOCATION-NEXT	Selects the next enabled location
;LOCATION-PREVIOUS	Selects the previous enabled location
;LOCATION-HOME	Selects the location flagged as the home location
;LOCATION-n	Selects the nth enabled location
;LOCATION-<name/ID>	Selects the location with specified name, or specified country ID (e.g. "UKXX0080", "London")
;LOCATION	Remain with the same location

Note that when a location command is encountered, all previous states are reset – i.e. Position reset to far left, selected forecast back to today.

Example

> *;IMAGE*
> *#Icon#*
> *0, 0*
>
> *;LOCATION-NEXT*
>
> *;IMAGE*
> *#Icon#*
> *0, 20*

The above will display the weather icon for the selected location, then the weather icon for the next enabled location.

Sub Views Objects

These objects are used to include the contents of another file

Syntax

> *;SUBVIEW*
> *<custom layout file>*
> *X,Y,W,H*
> *<optional touch command when object selected and activated – (I.e. tapped)>*

Example

> *;SUBVIEW*
> *somecustomlayoutfile.pcx*
> *0,40,100%,40*

The above will display the contents of the file "somecustomlayoutfile.pcx" at the location 0, 40, width 100%, height 40

Note that it takes the location context from any previous commands. So:

> *;SUBVIEW*
> *multiline-single.pwx*
> *0,35,W,30*
> *weatherconsole.exe #CITYID#*
>
> *;LOCATION-NEXT*
>
> *;SUBVIEW*
> *multiline-single.pwx*
> *0,60,W,30*
> *weatherconsole.exe #CITYID#*

the above will process the contents of multiline-single.pwx, then switch to the next location, and then display the contents of multiline-single.pwx again, which will be for the second location.

If the user taps on the region defined for the subview, then the application "weatherconsole.exe" will be executed with the cityid passed as a command line argument. The application is first searched within the PocketWeather main installation folder, then the "\Windows" folder, then the device's system paths (I.e. Lets the operating system try and find it.).

Refer to section on Touch objects for fuller details of the touch commands.

Touch Objects

These objects are used to include the contents of another file

Syntax

> *;TOUCH*
> *X,Y,W,H*
> *<optional touch command when object selected and activated – (I.e. tapped)>*

Example

> *;TOUCH*
> *0,35,W,30*
> *weatherconsole.exe #CITYID#*

If the user taps on the defined touch region, then the application "weatherconsole.exe" will be executed with the cityid passed as a command line argument. The application is first searched within the PocketWeather main installation folder, then the "\Windows" folder, then the device's system paths (I.e. Lets the operating system try and find it.).

The same text expansions as defined for the Text Objects are available.

Touch commands can also be used to cause the view to switch to another custom script layout file. For example:

> *;TOUCH*
> *0,35,W,30*
> *page1.pwc*

Note that the custom script layout file MUST be local to the current file's folder.

WeatherConsole Command Line Options

Valid WeatherConsole command line options are:

> *weatherconsole.exe <CITYID> <TABID> <SUBTABID>*

or

> *weatherconsole.exe <CITYID> <FULL PATH TO LAYOUT SCRIPT>*

where:

TABID	Function	SUBTABID
0	First Custom Layout	
1	METAR Map	0 – Map View 1 – Station Data View 2 – Station Compare
2	Location Forecast	0 – Summary 1 – Details 2 – Sun and Moon 3 - Hourly
3	Location Images	Image ID
4	Forecast Compare	0 – Forecast Summary 1 – Current Summary 2 – Current Details 3 – Temperature 4 – Precipitation 5 – Humidity 6 – Icons 7 – Wind Speed & Direction 8 – Daylight Times 9 – Trendings
5	Images Compare	
6	Earthquakes/World View	0 – World Map 1 - Details

When specifying the full path to a layout script, the first custom layout tab will always be selected. Note that the setting of the new custom layout script for the first custom layout tab is only temporary – navigating away and back will revert the first custom layout tab back to its configured script.

Conditional Statements

Syntax

>   ;IF-<CONDITION>-BEGIN
>   .
>   .
>   .
>   ;ELSE
>   .
>   .
>   .
>   ;END

or

>   ;IF-<CONDITION>
>   ;<OBJECT>

where <CONDITION> can be

Name	Description
FORECAST	If there is forecast weather data available
FORECAST-HOURLY	If there is hourly forecast weather data available
CURRENT	If there is valid and up-to-date current condition weather data available
METAR	If the current conditions data is from a METAR station
RADAR-n	If the selected location has a configured image icon "n" available
DISPLAY-APPARENT	If the apparent temperature is different from the actual temperature for the currently selected weather
DAY-DD	DD = Day of Week 0 = Sunday 1 = Monday 2 = Tuesday 3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday
DATE-YYYY-MM-DD	YYYY = year MM = Month DD = Day of Month
TIME-HH-MM-SS	HH = Hours (24-hour clock) MM = Minutes (optional) SS = Seconds (optional) ";IF-TIME-22" would trigger when the hour is 10pm
WINDGUST	If wind gusts are recorded
ALERT-FORECAST	If there is a weather forecast alert for the currently selected location
ALERT-CURRENT-CONDITIONS	If there is a current conditions alert for the currently selected location at any of the location's configured METAR stations
ALERT-CURRENT-CONDITIONS-n	If there is a current conditions alert for METAR station "n" for the currently selected location.
ALERT-n	If the "alertn" weather template feed variable has provided a string, where "n" is "0", "1"...
ALERT	If any of the "alert" variables provide a string
USER-n	If the "usern" weather template feed variable has provided a string,

Name	Description
	where "n" is "0", "1"...
USER	If any of the "user" variables provide a string
ANY-ALERT	If there is either a weather forecast, current conditions, or weather template feed variable.
DAYTIME	True if between sunrise and sunset
NIGHTTIME	True if between sunset and sunrise
TWILIGHT	Same as CIVILTWILIGHT
CIVILTWILIGHT	True if between civil twilight start and sunrise, or sunrise and civil twlight end
NAUTICALTWILIGHT	True if between nautical twilight start and sunrise, or sunrise and nautical twilight end
ASTRONOMICALTWILIGHT	True if between astronomical twilight start and sunrise, or sunrise and astronomical twilight end

Example:

> *;IF-CURRENT-BEGIN*
>
> *;TEXT*
> *#TEMP#*
> *0,45,40,15,c*
>
> *;ELSE*
>
> *;TEXT*
> *No Current condition data available*
> *0,45,40,15,c*
>
> *;ENDIF*
>
> or
>
> *;IF-RADAR-0*
>
> *;IMAGE*
> *#RADAR-0#*
> *40,0, 40, 57, cv*

Any of the above can be preceded with "NOT" to invert the condition. For example:

> *;IF-NOT-CURRENT-BEGIN*
>
> *;TEXT*
> *No Current condition data available*
> *0,45,40,15,c*
>
> *;ELSE*
>
> *;TEXT*
> *#TEMP#*
> *0,45,40,15,c*
>
> *;ENDIF*

Positioning

Syntax:

;CUSTOM
40,15

This allows you to set/reset the starting position to the specified X, Y position. This is needed when displaying forecast data using the ";FORECAST" object.

Note that the custom positioning is relative to previous positions, unless you specify "-ABSOLUTE", when the positions become absolute. Example:

;CUSTOM-ABSOLUTE
40,15

Displaying Forecast Data

Syntax:

> *;FORECAST*
> *<X-Offset>,<Y-Offset>,<IteractionCount>*
>
> *.*
> *.*
>
> *;FORECAST-END*

This causes it to iterate through the available forecast days, starting from the currently selected forecast day (default is todays), and repeating the objects/commands that follow, until it runs out of space, where the width of each day is <X-Offset>, and the height of each day is <Y-Offset>.

Specifying the optional <IteractionCount> causes the loop to stop after that many days.

The optional ";FORECAST-END" flags the end of the looping for each day. This allows for the script to handle multiple loops or lists of forecast data. If a subsequent ";FORECAST" command is declared, the selected forecast day continues from where the previous loop ended.

Note that for objects with display positioning, they automatically inherit the X-offset accumulated by the ";FORECAST" and ";CUSTOM" object declaration.

Example	Result
;FORECAST *40* *;TEXT* *#DAY#* *0,0,40,15,c* *;IMAGE* *#ICON* *0,9,40,40,cv* *;TEXT* *#HIGHLOW#* *0,45,40,15,c* *;TEXT-* *#RAIN#%* *0,54,40,15,c* *;FORECAST-END*	

Example	Result

```
;FORECAST
30,40,4

;TEXT
#DAY#
0,0,40,15,c

;IMAGE
#ICON
0,9,40,40,cv

;TEXT
#HIGHLOW#
0,45,40,15,c

;FORECAST-END

;CUSTOM-ABSOLUTE
80,0

;FORECAST
30,40,4

;TEXT
#DAY#
0,0,40,15,c

;IMAGE
#ICON
0,9,40,40,cv

;TEXT
#HIGHLOW#
0,45,40,15,c

;FORECAST-END
```

Displaying Forecast From Today

Syntax:

;CURRENT-FORECAST

This will select the current day's forecast context.

Example:

;CURRENT-FORECAST

;TEXT
#DAY#
0,0,40,15,c

Displaying Forecast From Tomorrow

Syntax:

> *;SKIP-FIRST-FORECAST*

When used in with ";FORECAST", this causes it to skip the current day's forecast and iterate through the remaining days.

Example:

> *;SKIP-FIRST-FORECAST*
>
> *;FORECAST*
> *40*
>
> *;TEXT*
> *#DAY#*
> *0,0,40,15,c*

Displaying Forecast For Specific Day

Syntax:

;FORECAST-n

This sets the forecast context to the "n"th day, where 0 is the current day, 1 is tomorrow, and so on.

Example:

;FORECAST-0

;IMAGE
#ICON
0,9,40,40,cv

;FORECAST-1

;IMAGE
#ICON
40,9,40,40,cv

;FORECAST-2

;IMAGE
#ICON
0,49,40,40,cv

;FORECAST-3

;IMAGE
#ICON
40,49,40,40,cv

Displaying Hourly Forecast

Syntax:

> *;FORECAST-HOURLY*
>
> *.*
>
> *.*
>
> *;FORECAST-END*

This causes it to iterate through the available hourly forecast, starting from the currently selected forecast, and repeating the objects/commands that follow, until it runs out of space, where the width of each hourly entry is the numeric value below – in this case "40".

The optional ";FORECAST-END" flags the end of the looping for each hour. This allows for the script to handle multiple loops or lists of forecast data. If a subsequent ";FORECAST-*HOURLY*" command is declared, the selected forecast hour continues from where the previous loop ended.

Example:

> *;FORECAST-HOURLY*
> *40*
>
> *;IMAGE*
> *#ICON*
> *0,9,40,40,cv*
>
> *;FORECAST-END*

To display a specific hourly forecast, use

> *;FORECAST-HOURLY-<n>*
> *40*

where n represents the number of hours to skip forecast. So ";FORECAST-HOURLY-24" skips forward 24 hours.

Displaying Current Conditions Data

Syntax:

> *;CURRENT*

This selects the current conditions context.

> *;CURRENT*
>
> *;IMAGE*
> *#ICON*
> *0,9,40,40,cv*