skin.ini Reference

General Notes

File format and location

The `skin.ini` file must be located in the `config` folder, along with the other ini files managed by the configuration software (e.g. `Aircast.ini`). You must create it manually using a text editor.

The file uses the typical syntax for .ini files:

```
[Section1]

Key1=Value

Key2=Value



[Section2]

Key1=Value

Key2=Value
```

The available section names and key names are given in the reference below.

Colors

All colors are specified in HTML color format, with six hex digits, e.g. `#FF0000`.

Fonts

For all fonts, there are four available settings:

`FontName`	Name of the font, e.g. `Arial`
`FontSize`	Size in pixels
`FontColor`	Font color in HTML format
`FontStyle`	1 = **bold**, 2 = *italic*, 4 = <u>underline</u>, 8 = ~~strikeout~~; add numbers to combine styles, e.g. 5 = **<u>bold+underline</u>**

In the reference below, we will only give the base name of the font setting, e.g. `TitleFont` – the available settings are then `TitleFontName`, `TitleFontSize`, `TitleFontColor` and `TitleFontStyle`.

Common Settings

Progress bar colors

The following settings can be used to change the appearence of the progress bar in the `Playlist`, `Player`, `Cartwall` and `ProgressBar` (for the global progress bar) sections.

`ProgressBarElapsedColor`	Elapsed section
`ProgressBarOutroColor`	Outro section
`ProgressBarRemainColor`	Remaining section
`ProgressBarRamp1Color`	Ramp1 remaining section
`ProgressBarRamp2Color`	Ramp2 remaining section
`ProgressBarRamp3Color`	Ramp3 remaining section
`ProgressBarElapsedRampColor`	Elapsed ramp section
`ProgressBarBorderColor`	Border around the progress bar and between parts in split mode
`ProgressBarIdleColor`	Fill color when player is empty
`ProgressBarIdleBorderColor`	Border when player is empty

Waveforms

`InactiveColor`	Background color when player is empty
`InactiveBorderColor`	Border color when player is empty
`BackgroundColor`	General background color
`NormalColor`	General waveform color
`FocusedColor`	General waveform color when waveform has focus
`CueColor`	Color of waveform before Cue In
`PlayedColor`	Color of elapsed part
`CueDataColor`	Color of cue data markers
`CueMarkerLineWidth`	Width of the cue position/marker lines
`CuePositionFont`	Font for cue position times (color is set automatically for cue markers)
`PositionColor`	Color of current playback position line
`PositionLineWidth`	Width of current playback position line
`EnvelopeColor`	Background color of volume envelope
`EnvelopeLineColor`	Line color of volume envelope
`EnvelopeBoxColor`	Color of volume envelope nodes/boxes
`EnvelopeLineWidth`	Width of volume envelope lines
`FadeLineWidth`	Width of fade triangle lines
`TimelineColor`	Color of timeline scale
`TimelineFont`	Font for timeline scale
`TimelineBackgroundColor`	Background color of timeline scale
`TitleFont`	Font for item title
`TitleBackgroundColor`	Background color for item title

Buttons

The following settings apply to all "graphical" buttons (not standard Windows buttons), e.g. the start/stop/…
buttons of the players, and toolbars built from these buttons (e.g the playlist control bar).

UseGradient	Use gradient fill? Default: `off`
`<State>Color`	The background color
`<State>GradientColor`	The gradient background color (if `UseGradient` is enabled)
`<State>BorderColor`	The border color
`<State>Font`	The font to use (for buttons that have text)

The available states are:

Active	Button is active
ActiveHover	Button is active, and mouse pointer is inside button
Inactive	Button is inactive
InactiveHover	Button is inactive, and mouse pointer is inside button
Disabled	Button is disabled

You can also leave out the `<State>` part in the rule name to adjust the settings for all states at the same time.

The button settings are prefixed by a particular button identifier as specified below,
e.g. `PlayButtonActiveFontName`.

Button toolbars

The following settings apply to all toolbars made of graphical buttons – currently only used for the Cartwall
toolbar. The individual buttons can be adjusted by prepending their `<ID>` to the particular settings.

Height	Height of the toolbar
FullWidth	Use all available horizontal space?
BackgroundColor	Background color
`<ID>ButtonVisible`	Set to `off` in order to hide the button
`<ID>ButtonMarginLeft`	Left margin in pixels
`<ID>ButtonMarginRight`	Right margin in pixels
`<ID>ButtonMarginTop`	Top margin in pixels
`<ID>ButtonMarginBottom`	Bottom margin in pixels
`<ID>ButtonCaption`	Button caption (for buttons that display a text)

Tree/list colors

All tree- and list-like components (e.g. the playlist, browsers, …) have a couple of common settings that can be
used in the appropriate settings:

BorderColor
DisabledColor
DropMarkColor

`DropTargetBorderColor`	
`FocusedSelectionBorderColor`	
`FocusedSelectionColor`	
`GridLineColor`	
`HeaderHotColor`	
`HotColor`	
`SelectionRectangleBlendColor`	
`SelectionRectangleBorderColor`	
`TreeLineColor`	
`UnfocusedSelectionBorderColor`	
`UnfocusedSelectionColor`	

Main Window

General settings

For the general settings, use the section `[MainWindow]`.

`Color`	General background color
`Background`	Full filename of a background image to use; works only in custom layout mode

Toolbar

The section `[Toolbar]` controls the appearance of the toolbar in the main playout window.

`Visible`	on or off (default: on)
`List`	In "List" mode, small icons are used, with the text next to them; default: off
`Captions`	Show button captions? default: on
`AutoSize`	Automatically adjust size? default: off

You can also hide the individual buttons in the toolbar (and the separators between them) by setting any of the following values to off:

`ShowNew`	*New* button
`ShowOpen`	*Open* button
`ShowSave`	*Save* button
`ShowSeparator0`	Separator between *Save* and *Insert*
`ShowInsert`	*Insert* button
`ShowProperties`	*Properties* button
`ShowDelete`	*Delete* button
`ShowSeparator1`	Separator between *Delete* and *Actions*
`ShowActions`	*Actions* button (if Menu Actions are defined)
`ShowEvents`	*Events* button

`ShowCartwall`	*Cartwall* button
`ShowMixEditor`	*Mix Editor* button
`ShowVoiceTracking`	*Voice Tracking* button
`ShowBrowser`	*Browser* button
`ShowSeparator2`	Separator between *Browser* and *Aircast*
`ShowAbout`	*Aircast* button

Playlist

Use section [`Playlist`] for settings that will effect all playlists, [`Playlist0`] only for the first playlist, [`Playlist1`] for the second, and so on.

Row appearance

The following settings can be used to change the appearance of the rows in the playlist:

`DefaultRowColor`	The default row background color (if no other entry matches)
`<Selector>RowColor`	The background color
`<Selector>RowPadding`	The row padding (extra space between text and cell boundary)
`<Selector>Font`	The font to be used

About selectors

Selectors allow you to specify settings that only affect particular rows, based on their type, state, etc. If you are familiar with HTML/CSS, this is pretty much like HTML classes and CSS class selectors.

For example, to change only the font of the "Title" column, use:

```
TitleFontName=Arial
```

Another example: Set the background to red for all items that are currently playing:

```
PlayingRowColor=#FF0000
```

You can also combine the various selector options – change the font style of the "Title" column for items that are currently playing:

```
PlayingTitleFontStyle=1
```

Generally, the selector for `RowColor` and `RowPadding` rules has the following form:

```
[Alternate]<Player State><Item Type><Item Class><Level>
```

For `Font` rules, there is an extra *Font Category* component:

```
[Alternate]<Player State><Item Type><Item Class><Font Category><Level>
```

The available options for the various components are described below. You can leave out any component, e.g. Player State, in which case the line will match **all** player states.

Remember that the skin.ini file is parsed from top to bottom. If there are multiple lines that match a particular row, the **last** line will have effect. So you should put more general rules to the top, and more specific rules to the bottom of the list.

Alternate

Rules starting with `Alternate` will only affect every other row – they allow you to create a striped table like this:

```
RowColor=#FFFFFF

AlternateRowColor=#F0F0F0
```

(Note that the – more general – `RowColor` rule comes first, it would override the `AlternateRowColor` rule otherwise.)

Player State

The following values can be used for Player State in order to select a row based on the state of the player in which the item is loaded:

Loading	Player is being loaded
Loaded	Player is loaded and ready
Next	Player is loaded and ready, and is next in sequence
Playing	Player is playing
Fading	Player is fading
EOF	Player is at EOF
Error	Player has an error, or prebuffering failed
Paused	Player is paused
PFL	Player is in PFL mode
Empty	Item is not loaded into any player
Played	Item has been played
Skipped	Item was skipped
Overflow	Item is "overflow" (will be skipped due to a following fixed time item)

There are two special "Player State" values that are used for rows that do not represent playlist items:

Underrun	An "Underrun" indicator node
Region	A region node (when you expand a Region Container playlist item)

Item Type

This component selects a row based on the Item Type as set in the item properties. Currently, the following values are available:

Unknown

Music

Voice

News

Weather

Traffic

Advertising

Package

Jingle

Sound

Effect

Trailer

Promo

Sponsorship

Sweeper

Drop

StationID

Bed

Instrumental

Show

Stream

Container

Playlist

Command

Break

```
Dummy

Silence

Error

Other

Custom1

Custom2

Custom3
```

Item Class

In addition to the Item Type (which the user can edit in the Properties dialog), you can also select rows based on the Class of the playlist item:

```
File

Stream

Break

Command

Container

Dummy

Other
```

Font Category

The following fonts can be adjusted through the skin:

PlayerName	The font used for the name of the player that the item is loaded into
Backtiming	The font used for the backtiming info displayed in the "Time" column
Title	The font for the title of the item
Artist	The font for the artist of the item
Duration	The font for the Duration column
Ramp	The font for the Ramp column
Ending	The font for the Ending column
Attribute	The font for
Comment	The font for expanded comments
Warning	Font used to display warnings, i.e. the Underrun warning etc.

Level

When the *Expandable Containers* GUI option is turned on, and you click the + button to expand a container, there will be a hierachy of playlist items, and the playlist will act like a tree. (And the container could have another container inside!)

The optional `Level` component in the selector allows you to define rules only for a particular level in that tree, e.g. to use a smaller font for the items inside the container. Level 0 is the top level, level 1 is for the items inside a container on top level, level 2 for items inside a container inside a container and so on.

DefaultRowColor vs. RowColor

`DefaultRowColor` and `RowColor` appear to do the same thing, as they define the color for all nodes that have no specific skin.ini setting (note that `RowColor` has an empty selector, so it matches **all** rows). But there are two differences:

1. `DefaultRowColor` only affects playlist item nodes, no "region" or "underrun" nodes.
2. When the GUI option /Item-specific colors have priority over skin.ini/ is turned off (so that skin has always priority over item-specific colors/, and you have a plain `RowColor` rule, that rule would always override the item-specific colors. Use `DefaultRowColor` instead in this situation (it has always lower priority than the item-specific color).

Overlay

The following settings control the position and appearence of the ramp/EOF overlay display:

`OverlayFont`	Font to be used for the overlay
`OverlayBackgroundColor`	Fixed background color; default: transparent
`OverlayOutlineWidth`	Width of the font outline
`OverlayOutlineColor`	Color of the font outline
`OverlayPosition`	Inside the playlist display, as cardinal direction: N, NE, E, SE, S, SW, W, NW. Default: C (center)
`OverlayPaddingX`	Horizontal padding
`OverlayPaddingY`	Vertical padding

Progress bar

Use the progress bar color settings described in the General section above.

Control bar/toolbar

To adjust the control bar (toolbar) of the playlist, use the following settings:

`ToolbarBackgroundColor`	Background color
`<Button>Button`	Appearance of the individual buttons, see *General* section above

The available buttons are:

`AssistAuto`	Combined ASSIST/AUTO button
`Assist`	ASSIST button
`Auto`	AUTO button
`Stop`	STOP button

Play	PLAY button
Next	NEXT button
Event	Box with time of next event
EndTime	End time box
HidePlayedItems	Hide Played Items switch
MoveUp	Move Up button
MoveDown	Move Down button
EditBar	Edit bar visibility switch
EditTiming	Timing button (edit bar)
EditHourlyFixTime	Hourly fixed time button (edit bar)
EditMoveUp	Move Up button (edit bar)
EditMoveDown	Move Down button (edit bar)

You can leave out the `<Button>` part to adjust all buttons at the same time.

Other settings

BackgroundColor	Background of the playlist where no rows are
SelectedFontColor	The font to be used for rows selected with the mouse
MainColumn	Index of the column where the +/- tree buttons should appear, first is 0, default: 2 (icon column)
GridLineStyle	`dotted` (default) or `solid`
ProgressBarHeight	Height of the progress bar
ProgressBarMargin	Number of pixels between progress bar and artist/title text
GridLines	Show horizontal lines between rows, `on` or `off` (default: `on`)
VertGridLines	Show vertical lines between columns, `on` or `off` (default: `off`)
HideSelection	don't show selected rows when playlist loses focus; `on` or `off` (default: `off`)
ColWidths	Widths of the individual columsin pixels, as comma-separated list of numbers; for the "auto size column", use *; hide a column with `0`
ColumnOrder	Order of the columns, as comma-separated list of numbers
TitlePercentage	How much of the width of the artist/title column should be used for the title in non-extended display? (default: 50)
IconSize	Fixed size of the playlist icons, in number of pixels; default: 0 (auto-size)

Player

Use the section `[Player]` to adjust the appearance of all players, `[Player0_0]` only for the first player of the first playlist (PLAYER 1-1, in terms of remote control), and so on.

Colors

The background color of the player can be adjusted with the following settings:

`<Player State>Color`	Background color for the particular state
`BorderColor`	Color for border outline

`Player State` can be left empty to adjust all colors at the same time, or use one of the following values:

`Empty`	Player is empty
`Loading`	Player is being loaded
`Loaded`	Player is loaded and ready
`Next`	Player is loaded and ready, and is next in sequence
`Playing`	Player is playing
`Fading`	Player is fading
`EOF`	Player is at EOF
`Error`	Player has an error
`Paused`	Player is paused
`Stopped`	Player is stopped (not used normally)
`PFL`	Player is in PFL mode
`FlashEOF`	Color used to indicate EOF warning

Fonts

The fonts for the individual states can be adjusted as follows:

`<Player State><Font Category>Font`

where `<Font Category>` is one of the following (or empty to adjust all fonts at the same time):

`Name`	Name of the player
`Time`	Time display
`Title`	Item title
`Artist`	Item artist
`State`	Player state (e.g. *CUE*, *ON AIR*)

For `<Player State>` use one of the values from the table in the *Colors* section above, or leave empty to adjust all states at the same time.

Progress bar

Use the progress bar color settings described in the General section above.

Waveform

Use the waveform settings described in the General section above, prefix all settings with `Waveform`, e.g. `WaveformInactiveColor=`….

Buttons

`ButtonSize`	Size of the buttons in pixels

`<Button>`Button	Settings for particular button

See the list in the General section for the available button settings.

`<Button>` can be one of the following (or empty to adjust all buttons at the same time):

Start	
Stop	
Pause	
Close	
PFL	
Loop	
Hook	
FadeOut	
Reset	

Other settings

`ProgressBarHeight`	Height of the progress bar
`ProgressBarMargin`	Margin above progress bar
`WaveformHeight`	Height of the waveform
`WaveformMargin`	Margin above waveform
`VolumeSliderWidth`	Width of volume slider
`VolumeSliderThumbWidth`	Width of volume slider thumb (default: `VolumeSliderWidth - 15`)

XML Layout

(todo)

Cartwall

Cartwall settings go into the `[Cartwall]` section.

General color settings

`ErrorColor`	Color of players in ERROR state
`PlayerDarkFontColor`	Used in STOPPED, PAUSED, ERROR state
`InverseFontColorThreshold`	The brightness value (0..255) below which the background color is considered "dark" (see next setting), default: 128
`PlayerInverseFontColor`	Used when current background color is darker than `InverseFontColorThreshold`
`ResizeNodeActiveColor`	Color of active resize nodes in design mode
`ResizeNodeInactiveColor`	Color of inactive resize nodes in design mode
`WindowBackgroundColor`	Background color for the Cartwall window

On Air Mode dependent color settings

The following colors can be adjusted based on the current "on air" mode:

`<Mode>PlayerColor`	Default player color if no specific color has been set for that cart
`<Mode>BackgroundColor`	Background color for cartwall
`<Mode>GridColor`	Grid line color
`InactiveBrightness`	Brightness adjustment for inactive players, default: 0.6

where `<Mode>` is empty (adjust all modes at the same time) or one of the following:

`OnAir`	ON AIR mode
`OffAir`	OFF AIR mode
`PFL`	PFL mode

Fonts

`IndexFont`	Font for the cart index/number
`TitleFont`	Font for the item title and artist
`TimeFont`	Font for the time display

Progress bar

Use the settings described in the General section of this document,

Toolbar

The cartwall toolbar can be adjusted through the `[CartwallToolbar]` section. For the settings, see section *Button toolbars* above. The IDs of the available buttons/elements in the toolbar are:

```
Menu

Favorites

NewPage

OpenPage

SavePage

DesignMode

Color

Waveform

OnAir

PFL

Next
```

Other settings

`IconMargin`	default: 4
`TabFont`	Font used for the tabs of the individual pages
`TabHeight`	Height of the tabs, set to 0 for auto height according to font
`TabWidth`	Width of the tabs, default 200, set to 0 for auto width

Browser

Unless otherwise noted, all settings go into the `[Browser]` section.

General Settings

`WindowBackgroundColor`	The general background color for the browser window
`BackgroundColor`	List background
`Font`	The font to be used
`NodeHeight`	The height of the list entries; should be adjusted when `FontSize` is changed

You can also adjust the tree/list colors as described in the General section.

The following settings have only effect for the "new-style" browser window:

`ButtonColorFrom`	
`ButtonColorTo`	
`ButtonHotColorFrom`	
`ButtonHotColorTo`	
`ButtonSelectedColorFrom`	
`ButtonSelectedColorTo`	
`ButtonSeparatorColor`	
`SplitterColorFrom`	
`SplitterColorTo`	
`DividerColorFrom`	
`DividerColorTo`	
`HeaderColorFrom`	
`HeaderColorTo`	
`FrameColor`	
`ToolPanelColorFrom`	
`ToolPanelColorTo`	
`ToolPanelHeaderColorFrom`	
`ToolPanelHeaderColorTo`	

Toolbar

The settings for the browser toolbar go into the `[BrowserToolbar]` section.

`Visible`	Set to `off` to hide toolbar
`List`	"List" mode (small icons, text next to icon)
`Captions`	Set to `off` to hide button captions
`AutoSize`	Set to `on` to make buttons auto size
`ShowAdd`	Set to `off` to hide /Add/ button
`ShowRefresh`	Set to `off` to hide /Refresh/ button
`ShowClose`	Set to `off` to hide /Close/ button

Comment Viewer

The *Comment Viewer* browser has two additional settings that go into the [`CommentViewerBrowser`] section:

`Font`	The font to use
`BackgroundColor`	The background color to use

Database Window

For the Library browser and Playlist view in the AircastDB window, there are

`Font`	The font to use
`NodeHeight`	Node height – increase when using a bigger `FontSize`

The settings go either into the [`DatabaseLibrary`] (Library list only), [`DatabaseTree`] (Library tree), [`LibraryPlaylist`] (playlist) or [`Database`] (all three) section.

Screen Objects

Some of the custom screen objects have additional skin settings.

Encoder Status

Section name for this settings is [`EncoderStatus`].

`MicButton`	Settings for *MIC* button (see General section)
`LineButton`	Settings for *LINE* button (see General section)
`BackgroundColor`	Background color to use

Mix Editor

The Mix Editor will respect all color settings in the [`Waveform`] section. See section on Waveform above. There are some extra color settings for the Mix Editor:

`WaveformEmptySpaceColor`	(where no waveforms are)
`WaveformTitleDividerColor`	line below titles
`WaveformTimelineDividerColor`	line below timeline
`WaveformWaveformDividerColor`	line in between waveforms

Mix-Editor-specific color settings can be added to the [`MixEditor`] section:

```
[MixEditor]

WaveformInactiveColor=#000000
```

The Mix Editor in v6.0 and above allows editing of nested playlist items (containers). You can use separate settings for nested items by specifying the level number (first level has index 0):

```
[MixEditor]

Waveform0InactiveColor=#000000

Waveform1InactiveColor=#0F0F0F
```