CasaTunes RTI Driver Cheat Sheet

The CasaTunes driver has changed a lot since the first version, with many powerful functions and variables that you can use. However, we have been hesitant to reorganize the functions and variables due to the risk of messing up dealers' current UI designs.

See How This Cheat Sheet Works if you need help.

NOW PLAYING 5

Metadata 5

[Shown in all UI examples, in Now Playing page] 5

[Variables] 5

Cover Art 5

[Shown in all UI examples, in Now Playing page] 5

[Variables] 5

Transport Buttons 5

[Shown in all UI examples, in Now Playing page] 5

[Commands] 5

[Variables] 5

Progress Bar 5

[Shown in all UI examples, in Now Playing page] 5

[Commands] 5

[Variables] 5

Shuffle, Repeat Buttons 5

[Shown in all UI examples, in Now Playing page] 5

[Example] 6

[Command] 6

[Variables] 6

[Visible State Variable] 6

QUEUE LIST AND QUEUE BUTTONS 7

Queue List 7

[Shown in all UI examples, in Now Playing page] 7

[Commands] 7

Play Queue Item 7

[Shown in all UI examples, in Now Playing page] 7

[Commands] 7

[Variables] 7

Remove Queue Item 7
[Shown in all UI examples, in Now Playing page] 7
[Commands] 7
[Variables] 7
BROWSING FOR MUSIC 8
Browse List and Browse Back and Home Buttons 8
[Shown in all UI examples, typically in Select Music page] 8
[Detailed Description] 8
[Commands] 8
[Command Parameters] 8
[Text Variables] 8
Play All and Add All buttons 9
[Shown in all UI examples, typically in Select Music page] 9
[Example] 9
[Commands] 9
[Visible State Variables] 9
Play Now, Add To Queue, Add & Play and Cancel buttons 9
[Shown in most UI examples, typically in Select Music page] 9
[Example] 9
[Commands] 9
[Visible State Variables] 9
Play Now Warning with confirmation buttons 10
[Shown in all UI examples, typically in Select Music page] 10
[Commands] 10
[Visible State Variables] 10
SEARCHING FOR MUSIC 11
Search Button

..... 11
[Shown in CasaTunes-iPad.rti, iPad and CasaTunes.rti, VP UI examples, in Select Music page] 11
[Example] 11
[Commands] 11
[Boolean State Variable] 11
SELECTING ROOMS AND SOURCES 12
Room Pulldown List 12

[Shown in CasaTunes-iPad, iPad Retina UI, in Now Playing page] 12
[Example] 12
[Commands] 12
[Variables] 12
Source Pulldown List 12
[Shown in CasaTunes-iPad, iPad Retina UI, in Now Playing page] 12
[Example] 12
[Commands] 12
[Variables] 13
Rotate Through Rooms 13
[Shown in CasaTunes, VP UI, in Now Playing page] 13
[Example] 13
[Commands] 13
[Variables] 13
Rotate Through Sources (Streams) 13
[Shown in CasaTunes, VP UI, in Now Playing page] 13
[Example] 13
[Commands] 13
[Variables] 13
Room and Source Lists 14
[Shown in CasaTunes, VP UI, in Room & Source Lists page] 14
KISS TOOLS 15
Browse in a Music Service 15
[Shown in CasaTunes.rti, VP UI example, in Direct Music Service Selection page] 15
[Example] 15
[Commands] 15
[Miscellaneous] 15
Jump Directly to Music Service Sub-menu 15
[Shown in CasaTunes.rti, VP UI example, in Direct Music Service Selection page] 15
[Example] 15
[Commands] 15
[Miscellaneous] 16
Favorites Buttons 16
[Shown in CasaTunes.rti, VP UI example, in Favorites page] 16

[Example] 16
[Commands] 16
[Miscellaneous] 16
Add and Delete CasaTunes Playlists (Favorites) 16
[Shown in CasaTunes.rti, VP UI example, in Now Playing and Select Music pages] 16
[Example] 16
[Commands] 17
[Variables and Miscellaneous] 17
HOW THIS CHEAT SHEET WORKS 18

NOW PLAYING

Metadata

[Shown in all UI examples, in Now Playing page]

Displays the song title, artist, album, genres. When playing music from Music Services, the results may be different, depending on the service, but will always be in logical order.

[Variables]

Metadata Line 0, Metadata Line 1, Metadata Line 2 and **Metadata Line 3,** found in **Now Playing Info.**

Cover Art

[Shown in all UI examples, in Now Playing page]

Displays image of Cover Art or similar image passed to CasaTunes by a Music Service.

[Variables]

Cover Art, found in **Now Playing Info**

Transport Buttons

[Shown in all UI examples, in Now Playing page]

Control Play, Pause, Next, Previous, Stop, Favorites, etc.

[Commands]

Various, found in **Stream Transport Controls.**

[Variables]

Important: use visible state variables as shown in the UIs, They are found in **Now Playing Visibility.** Not all songs/stations can be controlled by all transport buttons. CasaTunes will use the visible variables to display appropriate buttons.

Progress Bar

[Shown in all UI examples, in Now Playing page]

Displays progress of playing song. Allows repositioning song.

[Commands]

Song Position (0-100), found in **Stream Transport Controls.**

[Variables]

Progress for the Bar Graph Object, found in **Now Playing Info. Show Progress,** found in **Now Playing Visibility,** allows bar graph to be shown only when appropriate.

Shuffle, Repeat Buttons

[Shown in all UI examples, in Now Playing page]

Used to control how the queue is played.

[Example]

The **Shuffle Button** can be on or off, determined by the **Shuffle** variable. When On, the music in the queue plays in a random order. When Off, the queue plays in the order it was added to the queue.
The **Repeat Button** controls whether the music plays through once and stops or continuously, determined by the **Shuffle Mode** variable. Off means it will play through once, Once means it will repeat once and On means it will play continuously.
The **Repeat Button** can also be assigned the ability to only have two states On and Off, where On plays the music continuously.

[Command]

Shuffle, Repeat (On/Off/Once) and **Repeat On/Off** are all found in **Current Source/Stream Shuffle/Repeat Functions.**

[Variables]

Repeat Mode and **Shuffle** are used by the button text fields for displaying the current status of the buttons and are found in **Now Playing Info.**

[Visible State Variable]

Show Repeat and **Show Shuffle** are used to hide the buttons when they are not used (listening to network music services, for example). **NOTE: do not use "Repeat Visibility" for this. Repeat Visibility** is not recommended as it can only be used to hide something when the **Repeat Mode** is being used, but the mode is currently Off.

QUEUE LIST AND QUEUE BUTTONS

Queue List

[Shown in all UI examples, in Now Playing page]

Displays currently playing music. See how Queue can be saved as a CasaTunes Playlist.

[Commands]

Select Item From Play Queue List, found in **Play Queue List.** Item List Object is **Play Queue List,** found in **Play Queue List.**

Play Queue Item

[Shown in all UI examples, in Now Playing page]

Allows customer to change the currently playing song to another song in the queue. When using the visible variable, it will only be shown when appropriate.

[Commands]

Play Selected Queue Item, found in **Play Queue List.**

[Variables]

Show Play/Remove Queue Item Buttons, found in **Play Queue List.**

Remove Queue Item

[Shown in all UI examples, in Now Playing page]

Allows customer to remove songs from the queue. When using the visible variable, it will only be shown when appropriate.

[Commands]

Remove Selected Item from Queue, found in **Play Queue List.**

[Variables]

Show Play/Remove Queue Item Buttons, found in **Play Queue List.**

BROWSING FOR MUSIC

Browse List, Browse Back Button and Browse Home Button

[Shown in all UI examples, typically in Select Music page]

The **Browse List** is used to display the results when browsing for music. The **Browse Back Button** and the **Browse Home Button** control the **Browse List**. When the customer is looking for music to play, they can "Browse" for music using the **Browse List.** The list displays music items, including Music Services, Songs and Collections of music, such as Albums. How it responds to customers selections depends on what type of item is selected.

[Detailed Description]

The **Browse List** will contain a list of items that the customer can select. What happens next depends on the type of item:

[Collections] – A music item that contains a collection of other music items. When the customer clicks on a Collection, the driver fills the **Browse List** with the items in the collection.

[Queue-able Items] – If an item is something that can be added to a playing queue, such as an individual song, clicking on it will place it in the Queue.

[Non Queue-able items] – Some items, like radio stations, cannot be put in a queue, since they never stop playing. When a customer clicks on these items they start playing right away.

The **Browse Back Button** will cause the **Browse List** to load the previous level of the list. The **Browse Home Button** takes the **Browse List** all the way to the top (or a predetermined level using a parameter in the command) of the list.

[Commands]

Found in **Music Browsing List.** Use **Select Browse Item From List** for the scrollable **Browse List** and set the Item List Object to **Music Browse List.** Use **Back Browsing Up One Level** for **Browse Back Button** and **Reset Browsing List** for the **Browse Home Button.**

[Command Parameters]

Both the **Browse Back Button** and the **Reset Browsing list** contain a "to which level?" parameter. This is useful for dealers that are trying to keep the interface as simple as possible. For a simple interface, you would use buttons to select the Music Service and then the **Music Browsing List** would only show the items in that Music Service. By setting the "to which level" parameter to 1, the **Browse Back Button** and the **Reset Browsing list** button would only go back to display the items in the Music Service and would not show all of the Music Services. Discussed further in "KISS Tools".

[Text Variables]

Found in **Music Browsing List.** Use **Music list Heading** to display were the customer is while browsing. This can be in an inactive text button on top of the **Browse Back Button.**

Play All and Add All buttons

[Shown in all UI examples, typically in Select Music page]

Used to play all of the music items that are playable in the current level of the browse list. **Play All** replaces the currently playing queue with all of the playable music items and **Add All** adds the playable items to the end of the currently playing queue.

[Example]

When browsing for local music, if you open a collection of songs, such as the songs in a music album, if the customer clicks on **Play All,** all of the songs in the album will be put in the queue and start playing. Clicking on **Add All** puts the songs at the end of a current queue.

[Commands]

Play All Displayed Items and **Add All Displayed Items to Q** found in **Music Browsing List.**

[Visible State Variables]

Use **Show Play All and Add All Buttons** for both buttons and any grouping background. Variable found in **Standard Browse Music Visibility.** Buttons will only display when there are playable items in the music browse list.

Play Now, Add To Queue, Add & Play and Cancel buttons

[Shown in most UI examples, typically in Select Music page]

Used to control how a single, playable, music item (song) is played. These buttons will only display if the customer clicks on a music item in the browse list that can be added to the queue. **Play Now** replaces the currently playing queue with the single playable music item and **Add to Queue** adds the playable item to the end of the currently playing queue. **Add & Play** adds the music item to the end of the current queue and start playing the added song. **Cancel** cancels the operation and hides the buttons.

[Example]

When browsing for local music, if you open a collection of songs, such as the songs in a music album, if the customer clicks on one of the songs, the buttons pop up and the customer can decide how to play that song.

Note: clicking on a single item that cannot be added to a queue, such as a radio station, will not bring up the buttons. Instead, the radio station will immediately start playing.

[Commands]

Found in **Music Browsing List. Play Selected Browse Item** for **Play Now. Add Selected Item to Q** for **Add to Queue. Add Item to Queue and Play Now** for **Add & Play. Cancel Selected Browse Item(s)** for **Cancel.**

[Visible State Variables]

Use **Show Selection Popup** for **Cancel** and any grouping background visibility, found in **Popup Browse Music Visibility.** The Visible State Variables for the other buttons are found in **Standard Browse Music Visibility.** Use **Show Play Now Button** for **Play Now.** Use **Show Add to Queue**

Buttons for both **Add to Queue** and **Add & Play.** Buttons will only display when a playable music item is selected in the music browse list.

Play Now Warning with confirmation buttons

[Shown in all UI examples, typically in Select Music page]

Note: only used if **Turn on Play Now Warning** is checked in **User messages and warnings** section of the CasaTunes driver setup. If you do not include the play now warning buttons, make sure you uncheck the **Turn on Play Now Warning** checkbox!
This warning was added because some customers were creating complex queues by adding various albums and songs to the queue and then accidentally clicking on **Play Now** and losing their work. FYI: Some customers do not like this extra step of having to confirm every time they want to play music.

[Commands]

Play Now Warning OK Button and **Play Now Warning CANCEL Button** found in **Music Browsing List.**

[Visible State Variables]

Found in **Music Browsing List. Both buttons** and any warning message you would like to create will use **Play Now Warning** visibility state variable. Buttons and warning message will only display when an item or items are about to replace what is already playing in the queue.

SEARCHING FOR MUSIC

Search Button

[Shown in CasaTunes-iPad.rti, iPad and CasaTunes.rti, VP UI examples, in Select Music page]

Note, requires a keyboard page. Used by customer to search for music by entering a song title, artist or album name in the keyboard. Displays "Search" or "Browse". Once the button is set up with the correct macro, when it displays "Search", clicking on it will start the Search process. When it displays "Browse", clicking on it causes the driver to go back to browsing for music.

NOTE: If the customer is already browsing in a specific service, the search only looks inside that particular service. For example, if the Browse List is displaying the items in TuneIn, searching will only search inside of the TuneIn service. If the Browse List is displaying all of the available services, searching will search across all services.

[Example]

See macro assigned to the **Search Button** in the Select Music page of the iPad example. When a customer is looking for music they go to the Select Music page. When they click on the **Search Button** on the Select Music page, the macro assigned to the button checks the state of the button. If it is "True" (meaning it is ready to Search), it goes to the Keyboard page. The macro also issues the **Search/Browse For Music** command which causes the driver to change the state of the **Search Button** to "False". After the customer types in a search term in the Keyboard page and clicks on Enter, the Keyboard goes back to the Previous Page (the Select Music page) and fills in the Browse List with the search results. The customer can use this information to choose music to play. Meanwhile, the driver changed the text on the **Search Button** to "Browse" so the customer can go back to browsing for music, instead of searching for an item. If the customer clicks on the **Search Button** at this time, the macro does not change to the Keyboard page, but simply issues the **Search/Browse For Music** command again. This time the driver changes the state of the button back to "True", changes the text back to "Search" and resets the Browse List.

[Commands]

Used in the macro to change the state of the driver between Search and Browse. The command is **Search/Browse For Music** and it is found in the **Keyboard** section of the driver.

[Boolean State Variable]

Used to toggle the state of the **Search Button** between Search and Browse. The variable is the **Search/Browse Button State** variable, found in **Music Browsing List**. In the button's text field assign "Search" to the true state and "Browse" to the false state.

SELECTING ROOMS AND SOURCES

Room Pulldown List

[Shown in CasaTunes-iPad, iPad Retina UI, in Now Playing page]

Shows the currently selected room until the customer clicks on the room name. Then, the list of rooms displays and customer must select one to change the room that they are controlling.

[Example]

The List, Background and blue line below the Room Name all are assigned the **Show Room List** visible state variable so that they only display when the driver detects the **Make Room list Visible** command when the Room Name button is clicked. Once a room is selected from the list, the current room is changed and the list disappears.

[Commands]

Make Room List Visible, found in **Room Selection.** This command is assigned to the button that displays the current room name.
Select Room From List, found in **Room Selection.** This is for the pulldown list. Use **Room List,** found in **Room List** for the Item List Object.

[Variables]

Room Name, found in **Current Zone/Stream Info.** This is used for the Room Name button at the top of the list and will display the name of the current room. **Note: Do not use Room List Heading for the room name.**
Show Room List Visible State variable, found in **Standard List Visibility.**

Source Pulldown List

[Shown in CasaTunes-iPad, iPad Retina UI, in Now Playing page]

Shows the currently selected source or stream until the customer clicks on the source name. Then, the list of sources displays and customer must select one to change the source or stream that they are controlling.

[Example]

The List, Background and blue line below the Source/Stream Name all are assigned the **Show Source List** visible state variable so that they only display when the driver detects the **Make Source list Visible** command when the Source/Stream Name button is clicked. Once a source or stream is selected from the list, the current source is changed and the list disappears.

[Commands]

Make Source List Visible, found in **Change Source/Stream In Current Zone.** This command is assigned to the button that displays the current source or stream name.
Select Source From List, found in **Change Source/Stream In Current Zone.** This is for the pulldown list. Use **Source List,** found in **Source List** for the Item List Object.

[Variables]

Source/Stream Name, found in **Current Zone/Stream Info.** This is used for the Source Name button at the top of the list and will display the name of the current room. **Note: Do not use Source List Heading for the source name.**
Show Source List Visible State variable, found in **Standard List Visibility.**

Rotate Through Rooms

[Shown in CasaTunes, VP UI, in Now Playing page]

Change the room one at a time, when button is clicked.

[Example]

When the customer clicks on the **Room Button** the currently controlled room changes to the next one or previous one (selectable parameter).

[Commands]

Next Room, found in **Room Selection.** Includes Up/Down parameter.

[Variables]

Room Name, found in **Current Zone/Stream Info.** This is used for the **Room Button** and will display the name of the current room.

Rotate Through Sources (Streams)

[Shown in CasaTunes, VP UI, in Now Playing page]

Change the source or stream one at a time, when button is clicked.

[Example]

When the customer clicks on the **Source Button** the currently controlled room switches to the next source or previous source (selectable parameter). VP UI example shows how to set up a macro to switch to a different UI when the source is switched to an external device, like a cable box. When setting up for switching UIs for external sources, you need the "Go to Ext Source 1 on VP", "Go to Ext Source 2 on VP" and "Go to Now Playing on VP" System Macros and the Events that call them, as shown in the VP example.

[Commands]

Next Source, found in **Change Source/Stream In Current Zone.** Includes Up/Down parameter.

[Variables]

Source/Stream Name, found in **Current Zone/Stream Info.** This is used for the **Source Button** and will display the name of the current source.

Room and Source Lists

[Shown in CasaTunes, VP UI, in Room & Source Lists page]

Standard lists, use instructions for Pulldown lists, above, take a look at the example.

KISS TOOLS

KISS Tools are used to create very simple (KISS is an old acronym for Keep It Simple, Stupid) user interfaces. Some may not be so easy to set up the first time, but once implemented will make life easy for the customer.

Browse in a Music Service

[Shown in CasaTunes.rti, VP UI example, in Direct Music Service Selection page]

Used by customer to use a button to start browsing for music in a Music Service. As a result of the selection, the Browse List will be filled with the Music Services' items.

[Example]

In the VP example, a button was created with "TuneIn" in the text field. The **Browse in Music Service** command selects the TuneIn service. The button also tells the UI to switch to the Select Music page, where the TuneIn items, such as "My Favorites", "Local Radio", "Music", "Talk", etc., will be displayed for the customer to select. In this case the "switch to level" parameter in the Browse Back and Home buttons could be set to 1, so the customer cannot browse back farther than the top of the Music Service.

[Commands]

Browse in Music Service command with parameter "Music Service", found in **Music Browsing List.** Requires all of the customer's Music Services to be enabled in CasaSetup and credentials supplied for any premium services, and to be checked in the **Music Services** section of the driver setup.

[Miscellaneous]

Assign a Page Jump to the Select Music page to display the Browse List.

Jump Directly to Music Service Sub-menu

[Shown in CasaTunes.rti, VP UI example, in Direct Music Service Selection page]

Used by customer to use a button to browse directly in a sub-menu of a Music Service. As a result of the selection, the Browse List will be filled with the items in the Music Services' sub-menu that is selected. The sub-menu can be any collection.

[Example]

In the VP example, a button was created with "Local Radio" in the text field. The macro includes a **Browse in Music Service** command to select the TuneIn service, a delay to give the driver time to find the items in the TuneIn service and a **Browse by Name at Current Level**, with a string parameter for entering "Local Radio". When button is pressed, the driver changes to TuneIn, selects Local Radio and displays the local radio stations in the Browse List.
This command can be nested to go down deeper in the menus, as long as the string parameter is a "collection" (list of stations, list of songs, etc.). Delays must be placed between each level.

[Commands]

Browse in Music Service with parameter "Music Service", found in **Music Browsing List** as described in Select Music Service, above. **Browse by Name at Current Level** with string parameter for the name, found in **Music Browsing List**.

[Miscellaneous]

Assign a Page Jump to the Select Music page to display the Browse List.

Favorites Buttons

[Shown in CasaTunes.rti, VP UI example, in Favorites page]

Allows customers to use a button to go directly to a favorite CasaTunes Playlist. CasaTunes Playlists can be a group of songs or an internet radio station. Buttons will automatically appear when new CasaTunes Playlists are created and disappear when CasaTunes Playlists are removed. For CasaTunes Playlist add and delete commands see Add and Delete CasaTunes Playlists. A maximum of 20 **Favorites Buttons** can be used. If there are more than 20 CasaTunes Playlists, the first 20 will be available.

[Example]

In the VP example, you will see 16 buttons. Each button is indexed from 0 to 15 in this example. All **Favorite Buttons** are assigned a text variable to display the name of the indexed playlist. They are assigned commands to play the indexed playlist and they are assigned Visible variables indexed so that they are only visible if there is a playlist. So, for example, to create the first **Favorites Button** in the VP example, the button's text is assigned a text variable **Name for Playlist, Index 0,** the button's Visible variable is assigned **Visibility for Playlist, Index 0,** and the button's Driver Command is assigned **Play Selected Playlist**, with the Select Playlist parameter set to Playlist 0.

[Commands]

Browse in Music Service with parameter "Music Service", found in **Music Browsing List** as described in Select Music Service, above. **Browse by Name at Current Level** with string parameter for the name, found in **Music Browsing List**.

[Miscellaneous]

Assign a Page Jump to the Select Music page to display the Browse List.

Add and Delete CasaTunes Playlists (Favorites)

[Shown in CasaTunes.rti, VP UI example, in Now Playing and Select Music pages]

Allows customers to add and delete CasaTunes Playlists of music or favorite network stations. Add CasaTunes Playlists requires a Keyboard.

[Example]

Adding Playlists - In the VP Now Playing page, below the Queue list, the "Save Q as CasaTunes Playlist" saves whatever is in the queue, changes the page to the Keyboard page, and allows the customer to name and save the CasaTunes Playlist. For example, the customer could browse into TuneIn, browse in "Local Radio" and select KUNC (a public radio station in Fort Collins, CO). Once it is playing, the customer would use the Save Q as CasaTunes Playlist button to name the Playlist KUNC. Then, KUNC will show up when browsing in CasaTunes Playlists or when using the Favorites Buttons.

Deleting Playlists - While browsing through the list of CasaTunes Playlists, the customer can click on a playlist and then click on the **Delete Playlist** button shown in the Select Music page. The button only is visible when a CasaTunes Playlist is selected.

[Commands]

Save Queue as CasaTunes Playlist, found in **Play Queue List. Remove Playlist**, found in **CasaTunes Playlists.**

[Variables and Miscellaneous]

Show Remove CasaTunes Playlist Button visible state variable, found in **CasaTunes Playlist Button Visibility.**

Page Jump in the "Save Q as CasaTunes Playlist" button to jump to the Keyboard page to allow naming the CasaTunes Playlist.

HOW THIS CHEAT SHEET WORKS

Button or Function Title – The gold text is the title of the button or function.

[Shown in * UI example, in **** page] – tells you where to find the example of the button or function. The "UI example" will usually be a device in CasaTunes.rti, CasaTunes-iPad.rti, CasaTunes-Android.rti or CasaTunes-KX2.rti.**

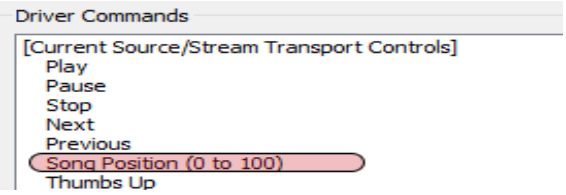
Includes a brief description of the button or function.

[Example]

A more detailed description of the button or function in an example that shows how the Button or Function works in UI.

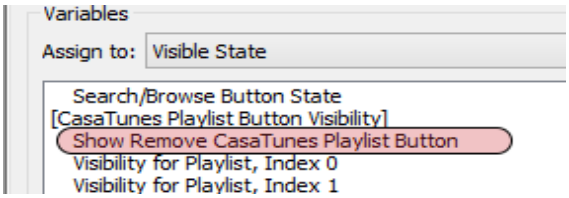
[Commands]

Where to find the command(s) that are used for this button, macro or function. For example, "**Song Position (0-100)**, found in **Stream Transport Controls**" means that the Song Position function used by the progress bar is found in the Stream Transport Controls:



[Visible State Variables], [Variables], [Boolean Variables]

Where to find the various Variables used in the UI design. For example, the "**Show Remove CasaTunes Playlist Button** visible state variable, found in **CasaTunes Playlist Button Visibility**", means you can find the variable that makes the **Remove Playlist** Button visible here:



[Miscellaneous]

Describes other misc. information that might be helpful.