**JumpStart Configuration File Documentation**

The JumpStartConfig.json file contains settings used by JumpStart to configure the initial generation of the new driver. Generally, the settings in this file would be consistent for a given environment and not need to change with each new driver generated.

The Json elements in the file are:

**Update**

This is an optional entry that can be used to specify the auto-update fields in the driver.xml file.

Possible sub-elements of Update are:

AutoUpdate

This is a Boolean string that specifies the value that should be included in the <auto\_update> tag included in the driver.xml file. If this is not included, the value will default to “True”.

ForceAutoUpdate

This is a Boolean string that specifies the value that should be included in the <force\_auto\_update> tag included in the driver.xml file. If this is not included, the value will default to “True”.

MinimumAutoUpdateVersion

This is an integer string that specifies the minimum version of the driver that will require an automatic update. It will be the value included in the <minimum\_auto\_update\_version> tag. If this is not included, the value will default to “1”.

MinimumOSVersion

This is a string that specifies the minimum version of the OS needed for the forced auto update. It will be the value included in the <minimum\_os\_version> tag. If this is not included, the value will default to “3.0.0”.

Example:

"Update": {

"AutoUpdate": "True",

"ForceAutoUpdate": "True",

"MinimumAutoUpdateVersion": "1",

"MinimumOSVersion": "3.0.0"

**Serial**

This is an optional entry that can be used to specify the default binding id and settings for any serial connections included in the driver.

Possible sub-elements for Serial are:

DefaultBindingID

This integer value specifies the binding id of the first serial connection defined in the driver. If this is not included, the value will default to 1. If a binding id is already in use, the value will be incremented until an unused value is found.

BaudRate

This is an integer that specifies the default baud rate listed in the <serialsettings> capability tag in driver.xml. If this is not included, the value will default to 9600.

DataBits

This is an integer that specifies the data bits value listed in the <serialsettings> capability tag in driver.xml. If this is not included, the value will default to 8.

StopBits

This is an integer that specifies the stop bits value listed in the <serialsettings> capability tag in driver.xml. If this is not included, the value will default to 1.

Parity

This is a string that specifies the parity value listed in the <serialsettings> capability tag in driver.xml. If this is not included, the value will default to “odd”.

FlowControl

This is a string that specifies the flow control value listed in the <serialsettings> capability tag in driver.xml. If this is not included, the value will default to “none”.

Example:

"Serial": {

"DefaultBindingID": 1,

"BaudRate": 19200,

"DataBits": 8,

"StopBits": 1,

"Parity": "even",

"FlowControl": "none"

},

**IR**

This is an optional entry that can be used to specify the default starting binding id for any IR connections included in the driver.

Possible sub-element for IR is:

DefaultBindingID

This integer value specifies the binding id of the first IR connection defined in the driver. If this is not included, the value will default to 1. If a binding id is already in use, the value will be incremented until an unused value is found.

Example:

"IR": {

"DefaultBindingID": 1

},

**Zigbee**

This is an optional entry that can be used to specify the default starting binding id for any Zigbee connections included in the driver.

Possible sub-element for Zigbee is:

DefaultBindingID

This integer value specifies the binding id of the first Zigbee connection defined in the driver. If this is not included, the value will default to 6001. If a binding id is already in use, the value will be incremented until an unused value is found.

Example:

"Zigbee": {

"DefaultBindingID": 6001

},

**ZWave**

This is an optional entry that can be used to specify the default starting binding id for any ZWave connections included in the driver.

Possible sub-element for ZWave is:

DefaultBindingID

This integer value specifies the binding id of the first ZWave connection defined in the driver. If this is not included, the value will default to 6001. If a binding id is already in use, the value will be incremented until an unused value is found.

Example:

"ZWave": {

"DefaultBindingID": 6001

},

**IP**

This is an optional entry that can be used to specify the default starting binding id for any IP connections included in the driver.

Possible sub-element for IP is:

DefaultBindingID

This integer value specifies the binding id of the first IP connection defined in the driver. If this is not included, the value will default to 6001. If a binding id is already in use, the value will be incremented until an unused value is found.

Example:

"IP": {

"DefaultBindingID": 6001

},

**Url**

This is an optional entry that can be used to specify the default starting binding id and settings for any URL connections included in the driver.

Possible sub-elements for URL are:

DefaultBindingID

This integer value specifies the binding id of the first URL connection defined in the driver. If this is not included, the value will default to 6001. If a binding id is already in use, the value will be incremented until an unused value is found.

DefaultHTTPPort

This integer value specifies the default value of the HTTPPort. If it is not included, the value will be 80.

DefaultRTSPPort

This integer value specifies the default value of the RTSPPort. If it is not included, the value will be 554.

DefaultAuthenticationRequired

This Boolean string specifies the default value for whether authentication is required. If it is not included, the value will be “True”.

DefaultAuthenticationType

This string specifies the default type of authentication. If it is not included, the value will be “BASIC”.

DefaultUserName

This string specifies the default username. If it is not included, the value will be “admin”.

DefaultPassword

This string specifies the default password. If it is not included, the value will be “pass”.

Example:

"Url": {

"DefaultBindingID": 6001,

"DefaultHTTPPort": 80,

"DefaultRTSPPort": 554,

"DefaultAuthenticationRequired": "True",

"DefaultAuthenticationType": "BASIC",

"DefaultUserName": "admin",

"DefaultPassword": "pass"

},

**Network**

This is an optional entry that can be used to specify the default starting binding id and settings for any Network connections included in the driver. Possible sub-elements for Network are:

DefaultBindingID

This integer value specifies the binding id of the first Network connection defined in the driver. If this is not included, the value will default to 6000. If a binding id is already in use, the value will be incremented until an unused value is found.

DefaultUserName

This string specifies the default username. If it is not included, the value will be “”.

DefaultPassword

This string specifies the default password. If it is not included, the value will be “”.

DefaultIPAddress

This string value specifies the default network address. If it is not included, the value will be “0.0.0.0”.

DefaultIPPort

This integer value specifies the default network port. If it is not included, the value will be 0.

DefaultBroadcastBindingID

This integer value specifies the binding id of the first Network broadcasting connection defined in the driver. If this is not included, the value will default to 6002. If a binding id is already in use, the value will be incremented until an unused value is found.

DefaultBroadcastPort

This integer value specifies the default broadcasting port. If it is not included, the value will be 9.

Example:

"Network": {

"DefaultBindingID": 6000,

"DefaultUserName": "admin",

"DefaultPassword": "pass",

"DefaultIPAddress": "127.0.0.1",

"DefaultIPPort": 80,

"DefaultBroadcastBindingID": 6002,

"DefaultBroadcastPort": 9

},

**MiniApp**

This is an optional entry that can be used to specify the default starting binding id for any MiniApp connections included in the driver. Possible sub-element for IP is:

DefaultBindingID

This integer value specifies the binding id of the first MiniApp connection defined in the driver. If this is not included, the value will default to 3100. If a binding id is already in use, the value will be incremented until an unused value is found.

Example:

"MiniApp": {

"DefaultBindingID": 3100

},

**AV**

This specifies the default classes that should be associated with different types of AV bindings that are specified in the driver. Possible sub-elements for AV are:

DefaultVideoInputClasses

Specifies an array of strings that should be used as the class types for any video input bindings listed in the driver. If not specified, the default value will be [ ].

DefaultVideoOutputClasses

Specifies an array of strings that should be used as the class types for any video output bindings listed in the driver. If not specified, the default value will be [ ].

DefaultAudioInputClasses

Specifies an array of strings that should be used as the class types for any audio input bindings listed in the driver. If not specified, the default value will be [ ].

DefaultAudioOutputClasses

Specifies an array of strings that should be used as the class types for any audio output bindings listed in the driver. If not specified, the default value will be [ ].

Example:

"AV": {

"DefaultVideoInputClasses": [

"COMPONENT",

"COMPOSITE",

"HDMI"

],

"DefaultVideoOutputClasses": [

"COMPONENT",

"COMPOSITE",

"HDMI"

],

"DefaultAudioInputClasses": [

"DIGITAL\_COAX",

"DIGITAL\_OPTICAL",

"STEREO"

],

"DefaultAudioOutputClasses": [

"SPEAKER",

"STEREO"

]

},

**Proxies**

This section specifies information for each of the supported proxies. Some Proxies will have some settings uniquely associated with them. They all have a list of the Proxy Capabilities with desired default values.

Possible sub-elements for individual Proxies include:

**AVSwitch Proxy**

The possible sub-elements for AVSwitch are: Capabilities. This lists the default capabilities that should be specified for this proxy.

Example:

"Proxies": {

"AVSwitch": {

"Capabilities": {

"can\_downclass": "True",

"can\_switch": "True",

"can\_upclass": "True",

"can\_switch\_separately": "False",

"requires\_separate\_switching": "False",

"has\_discrete\_balance\_control": "True",

"has\_discrete\_bass\_control": "True",

"has\_discrete\_input\_select": "True",

"has\_discrete\_loudness\_control": "True",

"has\_discrete\_mute\_control": "True",

"has\_discrete\_treble\_control": "True",

"has\_discrete\_volume\_control": "True",

"has\_toad\_input\_select": "True",

"has\_toggle\_loudness\_control": "True",

"has\_toggle\_mute\_control": "True",

"has\_up\_down\_balance\_control": "True",

"has\_up\_down\_bass\_control": "True",

"has\_up\_down\_treble\_control": "True",

"has\_up\_down\_volume\_control": "True",

"has\_video\_sense\_control": "True"

}

},

**Light Proxy**

Possible sub-elements for Light are:

ButtonLinkBindingIDBase

This integer value is the starting binding id for any button link bindings that may be associated with the light device.

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"Light": {

"ButtonLinkBindingIDBase": 300,

"Capabilities": {

"dimmer": "True",

"on\_off": "True",

"set\_level": "True",

"ramp\_level": "True",

"min\_max": "True",

"click\_rates": "True",

"click\_rate\_min": "250",

"hold\_rates": "True",

"hold\_rate\_min": "1000",

"cold\_start": "True",

"has\_leds": "True",

"supports\_broadcast\_scenes": "True",

"supports\_multichannel\_scenes": "False",

"hide\_proxy\_properties": "False",

"hide\_proxy\_events": "False",

"has\_button\_events": "True",

"advanced\_scene\_support": "True",

"load\_group\_support": "True",

"buttons\_are\_virtual": "False",

"has\_load": "True",

"max\_power": "0",

"min\_power": "-7"

}

},

**Lock Proxy**

The possible sub-elements for Lock are:

Capabilities:

This lists the default capabilities that should be specified for this proxy.

Example:

"Lock": {

"Capabilities": {

"is\_management\_only": "False",

"has\_admin\_code": "True",

"has\_schedule\_lockout": "True",

"has\_auto\_lock\_time": "True",

"auto\_lock\_time\_values": "0",

"auto\_lock\_time\_display\_values": "OFF",

"has\_log\_items\_count": "True",

"log\_item\_count\_values": "5",

"has\_lock\_modes": "False",

"lock\_modes\_values": "normal",

"has\_log\_failed\_attempts": "False",

"has\_wrong\_code\_attempts": "False",

"wrong\_code\_attempts\_values": "1",

"has\_shutout\_timer": "False",

"shutout\_timer\_values": "5",

"shutout\_timer\_display\_values": "5sec",

"has\_language": "False",

"language\_values": "English",

"has\_volume": "True",

"has\_one\_touch\_locking": "False",

"has\_daily\_schedule": "True",

"has\_date\_range\_schedule": "True",

"max\_users": "30",

"has\_settings": "True",

"has\_custom\_settings": "False",

"has\_internal\_history": "True",

"can\_edit\_user": "True",

"can\_edit\_user\_pin": "True",

"can\_add\_remove\_user": "True"

}

},

**MediaPlayer Proxy**

Possible sub-elements for MediaPlayer are:

HasAudio

This is a Boolean string value that specifies whether media player proxies support audio. If not specified, the value defaults to “True”.

HasVideo

This is a Boolean string value that specifies whether media player proxies support video. If not specified, the value defaults to “True”.

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"MediaPlayer": {

"HasAudio": "True",

"HasVideo": "True",

"Capabilities": {

"has\_http\_playback": "True",

"has\_discrete\_volume\_control": "True",

"has\_discrete\_mute\_control": "True"

}

},

**Receiver Proxy**

The possible sub-element for Receiver is:

Capabilities:

This lists the default capabilities that should be specified for this proxy.

Example:

"Receiver": {

"Capabilities": {

"has\_discrete\_volume\_control": "True",

"has\_up\_down\_volume\_control": "True",

"has\_discrete\_input\_select": "True",

"has\_toad\_input\_select": "True",

"has\_discrete\_surround\_mode\_select": "True",

"has\_toad\_surround\_mode\_select": "True",

"has\_discrete\_bass\_control": "True",

"has\_up\_down\_bass\_control": "True",

"has\_discrete\_treble\_control": "True",

"has\_up\_down\_treble\_control": "True",

"has\_discrete\_balance\_control": "True",

"has\_up\_down\_balance\_control": "True",

"has\_discrete\_loudness\_control": "True",

"has\_toggle\_loudness\_control": "True",

"has\_discrete\_mute\_control": "True",

"has\_toggle\_mute\_control": "True",

"surround\_modes": "<surround\_mode><id>22620</id><name>Stereo</name></surround\_mode><surround\_mode><id>22624</id><name>THX</name></surround\_mode>"

}

},

**SecurityPanel Proxy**

The possible sub-element for SecurityPanel is:

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"SecurityPanel": {

"Capabilities": {

"can\_set\_time": "True",

"can\_activate\_partitions": "True"

}

},

**SecurityPartition Proxy**

Possible sub-elements for SecurityPartition are:

Buttons

This is an arrow of elements for buttons A – D which may match the buttons on a security keypad. These buttons will show up on the Navigator interface. Each button element has values for:

tag: The XML tag label that will be in driver.xml associated with this button.

Visible: A Boolean string value that specifies whether this button is visible in Navigator.

Label: A string value that specifies how the button will show up in Navigator.

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"SecurityPartition": {

"Buttons": [

{ "tag": "button\_A", "visible": "True", "label": "Button A" },

{ "tag": "button\_B", "visible": "True", "label": "B" },

{ "tag": "button\_C", "visible": "True", "label": "C" },

{ "tag": "button\_D", "visible": "False", "label": "D" }

],

"Capabilities": {

"ui\_version": "2",

"supports\_virtual\_keypad": "True",

"has\_fire": "True",

"has\_medical": "False",

"has\_police": "False",

"has\_panic": "True",

"star\_label": "\*",

"pound\_label": "#",

"arm\_states": "Stay,Away",

"functions": ""

}

},

**Tuner Proxy**

Possible sub-elements for Tuner are:

HasVideo

This is a Boolean string value that specifies whether tuner proxies support video. If not specified, the value defaults to “False”.

HasAM

This is a Boolean string value that specifies whether tuner proxies support AM radio. If not specified, the value defaults to “True”.

HasFM

This is a Boolean string value that specifies whether tuner proxies support FM radio. If not specified, the value defaults to “True”.

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"Tuner": {

"HasVideo": "False",

"HasAM": "True",

"HasFM": "True",

"Capabilities": {

"selection\_delay": "0",

"has\_discrete\_input\_select": "True",

"has\_toad\_input\_select": "True",

"has\_tune\_up\_down": "True",

"has\_search\_up\_down": "True",

"has\_discrete\_preset": "True",

"has\_preset\_up\_down": "True",

"preset\_count": "3",

"has\_discrete\_channel\_select": "True",

"has\_channel\_up\_down": "True",

"preface\_band\_with\_tuner": "True",

"rf\_network\_type": "AMFM"

}

},

**TV Proxy**

The possible sub-element for TV is:

Capabilities

This lists the default capabilities that should be specified for this proxy.

Example:

"Tv": {

"Capabilities": {

"has\_discrete\_volume\_control": "True",

"has\_up\_down\_volume\_control": "True",

"has\_discrete\_input\_select": "True",

"has\_toad\_input\_select": "True",

"has\_discrete\_channel\_select": "True",

"has\_channel\_up\_down": "True",

"has\_discrete\_bass\_control": "True",

"has\_up\_down\_bass\_control": "True",

"has\_discrete\_treble\_control": "True",

"has\_up\_down\_treble\_control": "True",

"has\_discrete\_balance\_control": "True",

"has\_up\_down\_balance\_control": "True",

"has\_discrete\_loudness\_control": "True",

"has\_toggle\_loudness\_control": "True",

"has\_discrete\_mute\_control": "True",

"has\_toggle\_mute\_control": "True",

"has\_audio": "True",

"requires\_channel\_after\_input": "True"

}

}

}