

How to configure complex scenarios with Smart Control Applets in aixtream

Automation by the example of switching between multichannel sound and stereo based on the status of the center and surround channels of the input signal

In this blog post we will show the reader how Smart Control Applets in aixtream can be used to set up complex if-this-than-that scenarios. This is also sometimes referred to as “event-action programming.” One example of this could be a configuration that if a certain PIPE is stopped by an error, an email is immediately sent to one or more admins or technicians to inform them of the issue.

Here we will use an example brought to us by one of our customers to showcase how this works. They wanted to have one of their DVB radio programs switch automatically from 2-channel stereo to a 6-channel surround sound setting based on the incoming input signal. The idea is to detect activity on the 4 additional surround channels (center, sub-woofer, rear left, rear right) and if they reach a dedicated energy threshold switch the program automatically from stereo to surround sound. When the broadcast is over the program should also automatically switch back to stereo. This is not very easy to set up, as you can already guess by the length of this post, but it is an excellent example for the very high flexibility of our system and tells you everything you need to know so that you can configure similarly complex scenarios yourself.

To reach our goal we need three different parts:

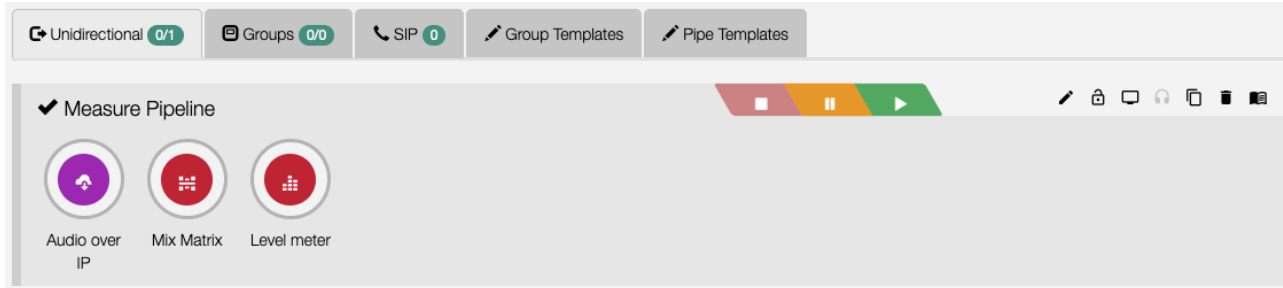
1. A measuring PIPE to check the activity of the surround channels.
2. Two presets, one for stereo, one for surround, in the target PIPE. These presets will be applied to the **Mix Matrix** element which will forward either only two or all six channels to the following PIPE elements.
3. Two **Applets** to switch from stereo to surround sound and back as defined in the presets.

We will now setup these parts step by step.

1. Configuring the measurement PIPE

1. PIPE setup

We start with setting up a new PIPE which contains an **Audio over IP** source, a **Mix Matrix**, and a **Level meter** element like this.



2. Audio over IP config

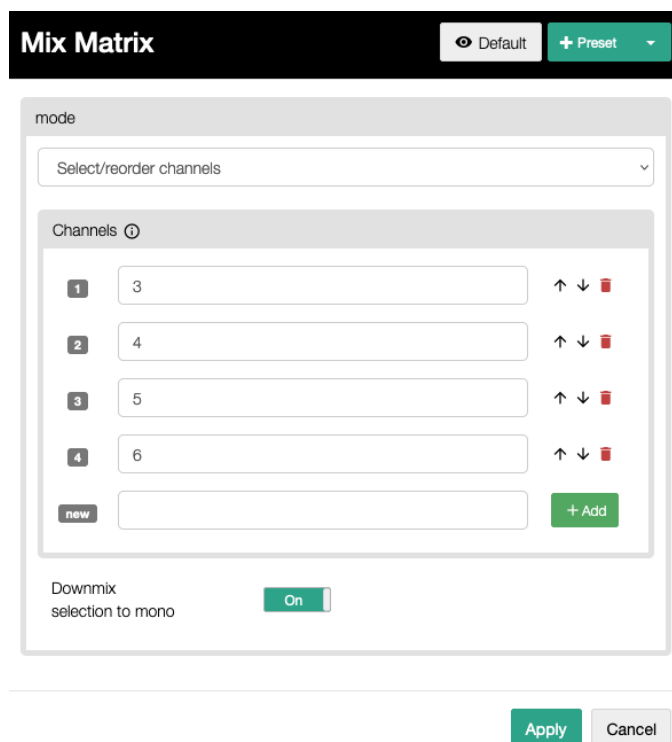
Next, we select a session of a multichannel AES67 source or paste a SDP (Session Description Protocol) file in the corresponding field in the **Audio over IP** source element configuration:

Audio over IP

Number of channels		<input type="text" value="2"/>
Sampling rate		<input type="text" value="48000 Hz"/>
Data available		<input checked="" type="checkbox" value="On"/>
Override display name		<input type="text"/>
Passthrough	<input type="checkbox"/>	<input type="checkbox" value="Off"/>
Session		<input type="text" value="AES67 Source DVB Multichannel Test / fernstation-282"/>
SDP	<input type="checkbox"/>	<input style="font-family: monospace; font-size: 0.8em;" type="text" value="v=0\no=- 3865048832 3865048832 IN IP4\n192.168.100.106"/>
Session name		<input type="text" value="AES67 Source DVB Multichannel Test / fernstation-282"/>
Size of jitter buffer		<input type="text" value="10"/> ms
Multicast interface	<input type="checkbox"/>	<input type="text" value="Default Static (en01)"/>

3. Mix Matrix config

In the **Mix Matrix** element, we select the “Select/reorder channels” mode and the channels we want to base our stereo-multichannel switching on. In this case we use the channels 3, 4, 5 and 6 (center, sub-woofer, rear left, rear right) of the 6-channel AES67 source. This way any activity on any single surround channel channels will be detected and a switching from stereo to multichannel will be triggered. The advantage of this is that the trigger also works for a 4.0 surround signal embedded into a 5.1 surround signal, or similar configuration, in which the center channel would be silent.



4. Level meter config

The setup of the **Level meter** is a bit more complex. First, we must activate the *Expert* mode in the top right corner of the configuration menu and ensure that the **Level meter** itself is enabled and that **Level Monitoring** is switched on. Now, on one hand, we have to set the parameter for switching from stereo to multichannel, based on audio activity on the surround channels being detected. On the other hand, the parameters for switching from multichannel to stereo based on the in/activity of the surround channels must be set.

Switch stereo to multichannel: The two parameters **Active Threshold** and **Hold Time** are important here. The **Active Threshold** is the energy level which has to be exceeded to trigger the applet. We suggest -60 dB as a good value here. In addition, we have to set the **Hold Time** which is the length of time the signal needs to be over the **Active Threshold** to trigger the applet. We set it to 2 seconds here since we want a relative fast switch from stereo to multichannel.

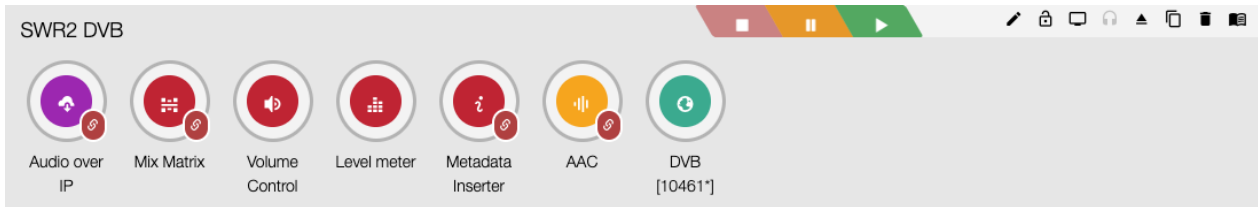
Switch multichannel to stereo: We need to set the parameters **Inactive Threshold** and **Warning Timeout** and **Error Timeout**. As with the case above the **Inactive Threshold** is the value which the energy level needs to fall below before it can be recognized as inactivity. We set it to -80 dB to allow for a kind of hysteresis to the **Active Threshold** above. The **Warning Timeout** and **Error Timeout** value is the counterpart to the **Hold Time**. We set it to 30 seconds since we want to have a stable multichannel setup and in cases in which the input is already set to stereo, we lose no information since the stereo audio is always included in the multichannel signal anyway

Level meter
Expert + Preset

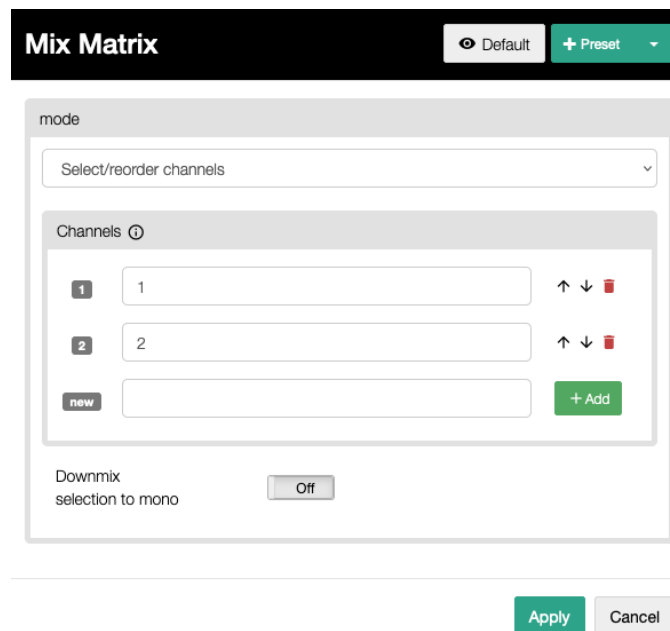
Measurement interval		<input type="text" value="500"/>	ms
Peak fall-off rate	ⓘ	<input type="text" value="10.0"/>	dB/sec
Peak hold time	ⓘ	<input type="text" value="300"/>	ms
Enabled		<input checked="" type="checkbox"/> On	
Level monitoring	ⓘ	<input checked="" type="checkbox"/> On	
Inactive threshold	ⓘ	<input type="text" value="-80"/>	dB
Active threshold	ⓘ	<input type="text" value="-60"/>	dB
Hold Time	ⓘ	<input type="text" value="2"/>	s
Warning Timeout	ⓘ	<input type="text" value="30"/>	s
Error Timeout	ⓘ	<input type="text" value="30"/>	s
Stop Pipe after error timeout	ⓘ	<input type="checkbox"/> Off	
Set alarm on timeout	ⓘ	<input type="checkbox"/> Off	

2. Creating presets for the Mix Matrix element in the DVB PIPE

Switching from stereo to multichannel and backwards requires switching the number of channels fed into the encoding chain of the DVB PIPE. To achieve this, we will use the **Mix Matrix** element. The PIPE could look like this:



The **Audio over IP** source should be configured like the source in the measurement PIPE from part 1. In the **Mix Matrix** element, we now need to create two different configurations, one for stereo and one for multichannel. We will define these as so-called **Presets**. We start with a stereo **Mix Matrix** which looks like this:



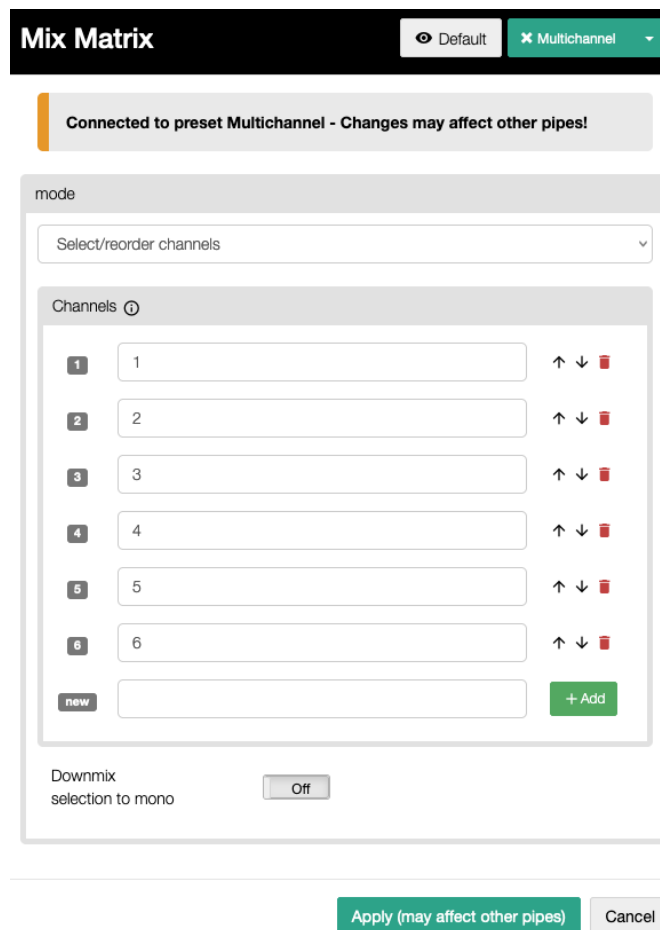
Now we must click on the **+** in right top corner in the **Preset** field. Then we type in the preset name "Stereo" and save it by clicking on the floppy disc symbol. The "Stereo" preset is created now.



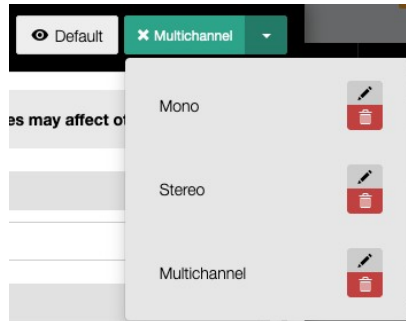
Next we click on the **x** next to "Stereo" to disable this preset.



Now we create the multichannel preset like the following image:



After that we can switch to the preset of the **Mix Matrix** element, we want to use, by clicking on the downward-pointing triangle at the top right corner of the config menu and then selecting the desired preset. Now everything is ready for the last step.



3. Creating applets for automated preset switching

An applet always consists of two parts: A trigger and an action. In our case the trigger is the measurement of the **Level meter** (or more specifically, the notification that there is audio in/activity) and the action is to switch to the desired preset in the **Mix Matrix** in the DVB PIPE. Unfortunately, as of today (July 2022), configuring the applets is rather intricate, but we plan to make it more intuitive in the future.

1. Prerequisites

First, we need to retrieve the UUIDs (Universally Unique Identifier) of some objects. These can be found in the Debug menu. There we can find all internal objects of aixtream including their UUIDs.

Activating the Debug menu

To activate the Debug menu in the aixtream GUI, please follow these steps:

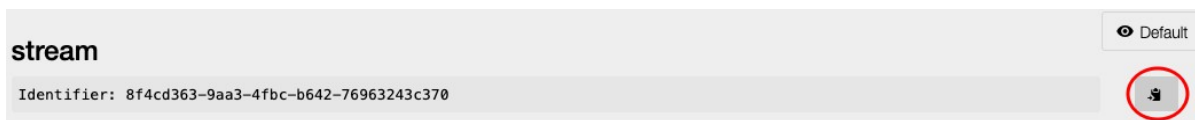
1. Go to **Settings (Gear icon) > System**
 2. Activate the **Expert** view there
 3. Scroll to the bottom of the menu and enable **Debug mode (WebRemote)**
 4. Click on Save Changes.
 5. Reload the Web GUI by pressing CTRL+R *or* clicking the reload symbol next to the URL in the browser.
- ➔ You can see a new menu tab with a wrench symbol (the **Debug** menu).

2. Creating the applet

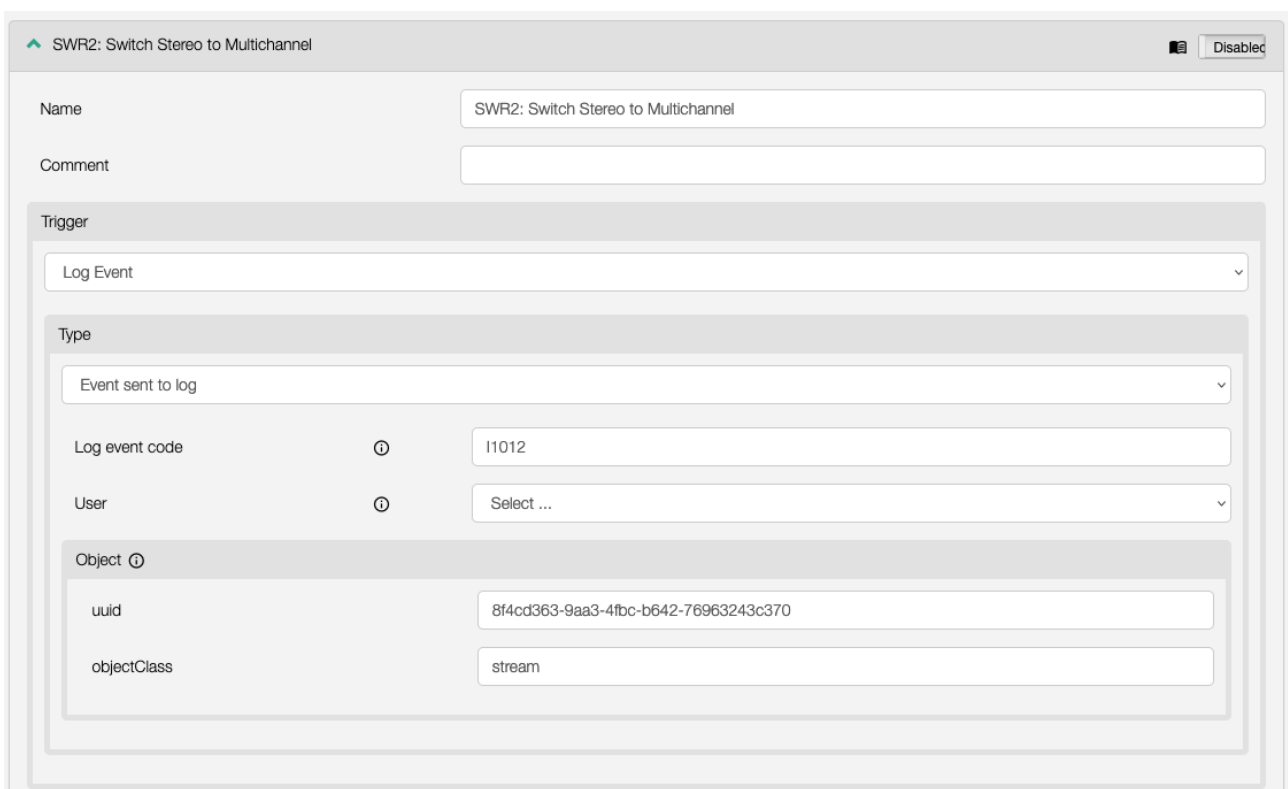
In the Applet menu, we now create a new applet and give it a memorable name. As trigger we select “Log Event” and set the **Event Code** to “11012”. This code will be logged if a level meter finds audio activity on the input (again). To get only the “11012” events of the measure PIPE we need to set the Object **objectClass** property to “stream” (which is the internal name for a PIPE) and the **Object UUID** to the UUID of the measure PIPE.

3. Identifying and inserting the UUID of the measurement PIPE

To find out the PIPE UUID, we go to the Debug Tab (wrench symbol). We recommend that you open the Debug Tab in a separate tab of your browser for the sake of convenience. Now we click on **stream** and then on the name of the measure PIPE. At the top you can now find the UUID of the selected measure PIPE which can be copied to the clipboard by clicking the button on the right.



Now go back to the applet we created earlier and insert the copied UUID to the **object UUID** field in the trigger. Afterwards, the trigger configuration should look similar to this:



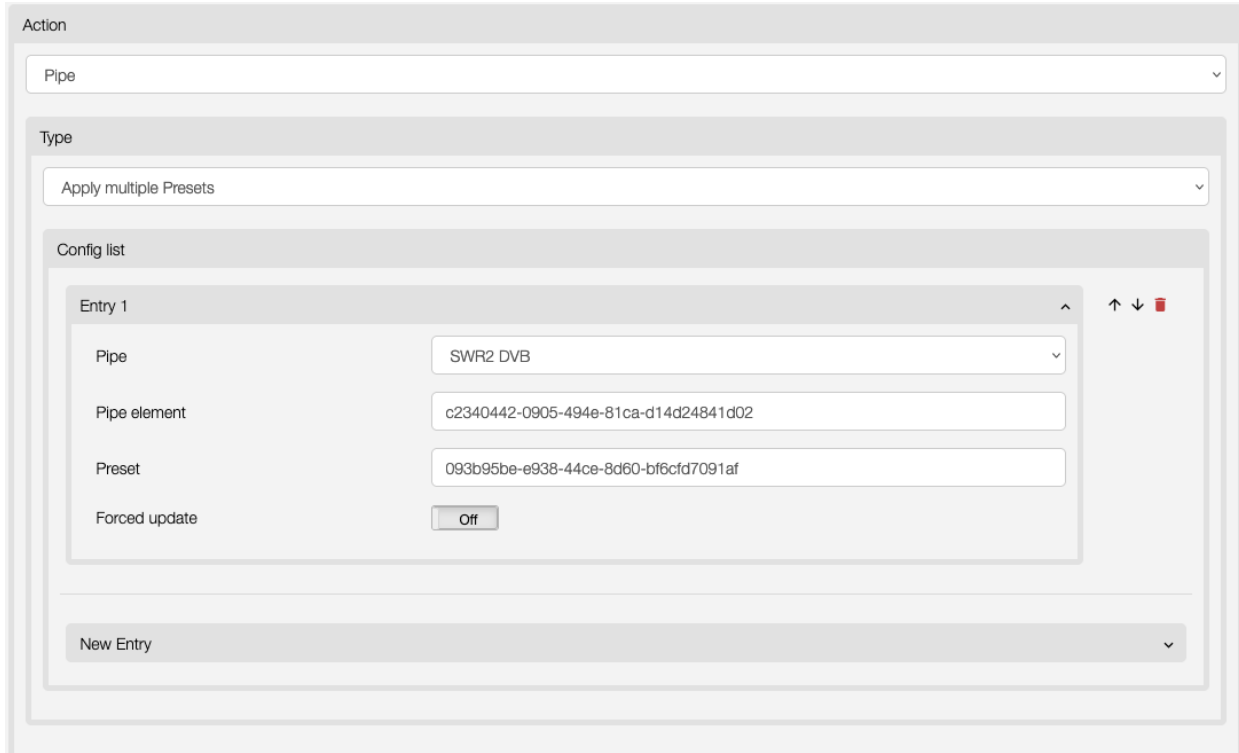
4. Setting the action

We are now ready for the action part of the applet. We select **Action** "Pipe" and the **Type** "Apply multiple Presets" from the respective dropdown menus. Next we open the **New Entry** by clicking on the "v" on the right and select the PIPE for which we want the preset to switch between stereo and multichannel.

5. Identifying and inserting the UUID of the stream PIPE and presets

We now have to insert the UUID of the "Pipe element" where the preset should be changed.

We find it by going to the Debug tab once more. We select the DVB stream for which we want to change the present, scroll down to **elements** and open **Entry 2** which is the **Mix Matrix** element of our example PIPE (make sure to confirm the placement of the elements in your PIPE accordingly). From here we copy the UUID and insert it into the **Pipe element** field of the applet action. In the field **Preset** we insert the UUID of the preset element we want to switch to — in this case the multichannel preset. Once again, we can obtain it by switching to the **Debug** menu and heading over to the **elementPreset** objects. From there we copy the UUID of the "Multichannel" preset, insert it into **Preset** in the applet action and click the **Add** button. Afterwards, we click **Save**.



The screenshot shows a configuration window for an "Action" applet. At the top, the "Action" dropdown is set to "Pipe". Below it, the "Type" dropdown is set to "Apply multiple Presets". A "Config list" section contains one entry, "Entry 1", which is expanded to show the following fields:

- Pipe:** A dropdown menu showing "SWR2 DVB".
- Pipe element:** A text input field containing the UUID "c2340442-0905-494e-81ca-d14d24841d02".
- Preset:** A text input field containing the UUID "093b95be-e938-44ce-8d60-bf6cfd7091af".
- Forced update:** A toggle switch set to "Off".

At the bottom of the configuration list, there is a "New Entry" button with a dropdown arrow on its right side.

6. Creating the applet for the reverse case

We now clone this applet and then change its name. We must replace the event code "I1012" with "W1012" in the trigger part. The W1012 event signals that the audio is inactive at the level meter. Alternatively, we could also use event E1012 here which would then use the **Error Timeout** option in the **Level meter** configuration instead of the **Warning Timeout**. Afterwards, we replace the **Preset** UUID in the action part of the applet by the UUID of the Stereo preset (once again found in the Debug menu).

It should then look similar to this:

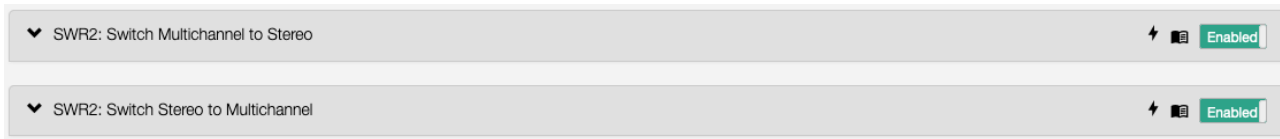
The screenshot shows the configuration interface for an applet titled "SWR2: Switch Multichannel to Stereo". The interface is divided into several sections:

- Name:** SWR2: Switch Multichannel to Stereo
- Comment:** (empty)
- Trigger:**
 - Log Event: Log Event
 - Type: Event sent to log
 - Log event code: W1012
 - User: Select ...
 - Object:
 - uuid: 8f4cd363-9aa3-4fbc-b642-76963243c370
 - objectClass: stream
- Action:**
 - Pipe: Pipe
 - Type: Apply multiple Presets
 - Config list:
 - Entry 1:
 - Pipe: SWR2 DVB
 - Pipe element: c2340442-0905-494e-81ca-d14d24841d02
 - Preset: fab342a4-711c-4b3a-8719-14750e0b0b7f
 - Forced update: Off
 - New Entry: (button)

At the bottom of the interface, there are three buttons: Save (green), Clone (orange), and Remove (red).

7. Activating the applets

At the end we have to activate the two applets by clicking the buttons on the top right.



Summary

We now have a fully automated system which switches a DVB PIPE from stereo to multichannel sound and back based on the audio activity of the surround channels of the input. The audio activity detection is realized with a measurement PIPE with a **Level meter** at the end, which can be extensively configured. If a radio station has a fixed schedule for multichannel programs, it would also be possible to use a **Date & Time** trigger in an applet to do the switching from stereo to multichannel.

The switching of presets in the applets is a convenient way to reconfigure elements in PIPEs. For example, it is not only possible to switch the number of channels but also to switch the bitrate of the encoder element the same way. Presets and preset switching are techniques that can improve the speed and quality of your aixtream operation in general.

The setup of the applets seems to be rather complex at first sight, but it opens a variety of application scenarios because it is so very flexible. A different example scenario would be to measure the activity of an input with a measurement PIPE and if there is no activity starting a separate PIPE to serve some audio from a file as a backup output.

These functions in combination offer a breadth of possibilities as well as flexibility. We are very excited to see our customers experiment and develop their own new ways of using them.