
Subject: Aux behaviour for Senderella project: Opinions requested

Posted by [drfrankencopter](#) on Sun, 21 Mar 2010 21:33:09 GMT

[View Forum Message](#) <> [Reply to Message](#)

Ok guys,

I've got Paris' pan law implemented into the version of Senderella that I'm working on, and am just about work on changing the volume control to be in dB...and that brings me to my request for your opinions.

What volume range would you like to see in the senderella plug-in? For reference, Paris' auxes go from -20dB to +20dB. This seems like a little too much in the positive sense. Do users actually make use of that positive gain? I was thinking of going with a similar range, but maybe more like -30dB to +10dB. What do folks think about that? Do you want more aux range? Less? When Senderella is instantiated as a return, the volume control will act as an effect send master volume just like those controls in Paris.

Realize that there is a tradeoff between range and resolution to be negotiated here.

Another thing to think about...do you want saturation on the aux buss? Should there be headroom limits, or go with typical native behaviour of full floating point resolution?

Any opinions on desired aux behaviour would be great,

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested

Posted by [kerryg](#) on Sun, 21 Mar 2010 22:18:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

-30 to +10 seems more useful than +/-20, though I could live with less range (if I want to do more profound gain trimming I'll do it elsewhere).

Saturation? What kind of saturation are we talking about here and what would it sound like? My first instinct is that there are many other ways of generating that if desired, but I guess it'd depend.

Maybe the conceptual key is making it behave like the rest of PARIS, ie "forgiving".

Subject: Re: Aux behaviour for Senderella project: Opinions requested

Posted by [drfrankencopter](#) on Mon, 22 Mar 2010 00:08:59 GMT

[View Forum Message](#) <> [Reply to Message](#)

kerryg wrote on Sun, 21 March 2010 15:18

Maybe the conceptual key is making it behave like the rest of PARIS, ie "forgiving".

EXACTLY... Paris lets you saturate in so many different places, I just thought that since this version of Senderella is just for us, maybe we'd want to replicate that ability to saturate the aux sends as well. I'll only treat it as a low priority item, as bug fixing is the top order of business.

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [Dimitrios](#) on Mon, 22 Mar 2010 21:15:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,
Have I missed something here ??
What do you mean by "Paris" senderella working on ??
Does someone work on a senderella version upgrade ?
Regards,
Dimitrios

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [kerryg](#) on Mon, 22 Mar 2010 21:57:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Dimitrios - yep, by permission of Senderella's developer, Kris is working on a version of Senderella more customized for PARIS use. First stop is getting the pan and gains in musically useful increments.

By the way - I was thinking of organizing your previous tutorials into the same sort of illustrated "step by step" tutorial shape as this for the Wiki, would that be OK with you?

Incidentally - Senderella isn't the only thing that's gone open source and is thus available for tweaking for PARIS-friendliness - so has Wormhole. So there's new and fun avenues to explore. We could use more coding eyes on all of this though, if there's folks out there that are interested.

- Kerry

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [drfrankencopter](#) on Tue, 23 Mar 2010 01:17:52 GMT

Hi Dimitrios,

It was your posts that inspired me to try Senderella, and sure enough it worked. But, there are little issues that bug me with it. So, I found a copy of the source code to an older version, and contacted the developer (ModuLR) and asked if he minded if I branched his code. ModuLR was more than happy to let me hack at his code, and here's what I'm working towards:

- 1) Send volume to be in dB instead of % (I never like working in % for audio)
- 2) Integrated pan knob that follows the Paris pan behaviour. I managed to get this working tonight (at least in Reaper....I haven't tested on my Paris rig yet). This means that it's no longer necessary to use 2 instances of senderella (set to different channels) per channel in order to get a panned signal.
- 3) Latency compensation. Here's my idea...on a return instance, there will be a slider that allows you to delay the direct pass-through path of the sends by the selected # of samples. This would let you insert say waves Ren-compressor on the return, and delay the sends to match its 64 sample latency.
- 4) Variable delay on each send instance...this could serve as a variable pre-delay for a reverb send. I've always wanted to have different pre-delays on different instruments in order to 'place' them in different spots in the reverb...and now it's possible.
- 5) Bug fixes...the source code is pretty messy in spots, and there is code in there that will never be executed. I understand the original was never really considered anything more than a hack.

If you've got some feature requests, let me know....

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [kerryg](#) on Tue, 23 Mar 2010 02:33:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

Heya Kris - is a wet/dry mix feasible?

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [drfrankencopter](#) on Wed, 24 Mar 2010 13:46:30 GMT

[View Forum Message](#) <> [Reply to Message](#)

Re: Wet/dry...

It might be possible, but I'm not sure how you think it would work from a user interface perspective. As it stands, wet dry is determined by the fader positions. You could get a true wet/dry implementation by using Paris 3.0 fader groups....that might be the best way to do it.

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [kerryg](#) on Wed, 24 Mar 2010 15:04:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

I posted that before I'd really grasped your latency compensation idea, which looks like a better way of achieving what I was thinking about - ways of making certain wet and dry signals remain sample locked...

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [Dimitrios](#) on Fri, 26 Mar 2010 12:30:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,
NICE !

One thing is that edstransfer should be 8,8 or 8,16 the latter working better with disk transfer on many audio tracks.

I suspect though that this edstransfer might overheat cards.

I had a couple burned out that might be of that cause but could not either...

I have also read from other posters that this eds transfer might overheat cards...

So better have plenty of ventilation to blow some good air right on the cards.

Senderella must be wrapped. Chainer was right so ffx4 as you say is ok too BUT ffx4 is DX so you have the 64 plugin limit !!

Chainer is vst so there is no 64 plugin limit there...

Regards,
Dimitrios

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [drfrankencopter](#) on Fri, 26 Mar 2010 16:34:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

Ok, I learned a few things last night.

1) I managed to get VSTMonitor to work. This is a plug-in that is provided with the VST SDK. I probably shouldn't distribute it, but it will help the community learn more about the Paris VST configuration. So, here's the download from my site (This link will not be left active for more than a couple of weeks).

<http://web.ncf.ca/fk824/paris/VSTMonitor.dll>

I'd ask that folks who're interested in getting Senderella working well in Paris install VST monitor in their system (it can be easily un-installed later by deleting the dll file). What I'm interested in seeing is the block size that different paris configuration settings use. The way to look for this is to insert the VST monitor plug-in on a channel, click on the 'Display' button, and look for an entry that is like PROCESSREPLACING(2048). Here, 2048 is the block size that paris is calling the VST plug-in with.

I'm curious to see if the EDSTRANSFER setting has something to do with the VST blocksize or not, or perhaps it's affected by Disk I/O size. I'd do this all myself, but since there's a re-start of paris required for each setting change it's a long process. If other folks can just run VSTmonitor on their setups, and tell me what the blocksize is, then it'll be faster (and I won't need to test extreme values).

2) I installed my built version of senderella into the Paris VST path, and it while it certainly did something, it was not as promising as the results were in Reaper (which I was testing on a different machine too). I was not getting stereo returns, and the audio was 'stuttering'. My current suspicion is that this is a multi-core problem, as the Senderella source code puts no protection on the shared data between plug-in instances, and relies on the Host to determine the order of processing. So, you can conceivably run into a situation where if the load gets split between multiple cores, that the returns get processed before the sends, and the buffers are empty. I'll need to do more analysis to see if this is the case.

3) Paris VST does not call the 'process()' VST function, but rather calls 'processReplacing()'....I found this out by setting breakpoints in the code, and found that no breakpoints in the 'process()' function were ever executed. Getting breakpoints enabled in the code executing within paris was a bit of a breakthrough...but the data contained in the shared memory buffer (the data shared between Senderella instances), did not look like what I expected, nor did it look like what it did when I tested in Reaper. This behaviour may be different when wrapped...I'm not sure, and it will need to be tested.

4) My dx-vst wrapper expired...I bought FXpansion 3.0 eons ago, but lost the activation code. I understand it's free now, does anyone have the link?

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested

Posted by [dnafe](#) on Fri, 26 Mar 2010 21:03:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Is that vdx33b?

if so email me at dnafe@rogers.com

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [drfrankencopter](#) on Sat, 27 Mar 2010 14:51:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Learned a bit more about Paris.

Disk I/O, and EDSTRANSFER does not appear to affect the block size (buffer) sent to VST plugins. It appears that the plugin transfer block is fixed at 2048 samples.

Cheers

Kris

PS: My version of senderella also behaves completely wrong under Paris VST. I'll soon be checking it under a DX wrapper.

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [drfrankencopter](#) on Mon, 29 Mar 2010 04:02:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

Learned a whole ton about Paris' native FX oddness this weekend.

1) Paris tells VST plugins that it is using a 1024 sample block size (comparable to an ASIO buffer size, it's the size of the data 'chunks' that Paris passes to the plug-ins). BUT, Paris actually sends 2048 as the number of samples to process. What's even more strange is that if you look at the actual samples that are sent to be processed, it's only the 1st 2000 that actually contain data, and the final 47 are all zeros

2) I ran into all sorts of problems running a send and return on the same submix. I was running the send on channel 1, and the return on channel 3/4 and I was getting stuttering sort of sounds. I fought with these sounds all weekend, and made no headway. But, out of desperation I tried moving the returns to a native submix, and VOILA....the problem rectified itself, and the return audio sounded just fine. My best guess is that there is an order of operations issue that needs to be worked out, and the moving the returns to the native submixes fixes it as they are processed last.

3) The plugin behaved very differently when wrapped by FXpansion versus under Paris' native VST host. The paris implementation was unpredictable at best.

Cheers

Kris

Subject: Re: Aux behaviour for Senderella project: Opinions requested
Posted by [justcosmic](#) on Sun, 22 Dec 2013 16:16:46 GMT

I have been looking at using senderella to send audio in/out of Paris to/from Reaper. For me the "%" instead of "DB" is much more useful for being able to get unity gain (i.e. 100% on the original senderella). So having the option would be helpful. It is also very difficult to dial in precise db values with just a slider.

In my initial tests from one channel to another within reaper, the original senderella does indeed exhibit buffer noise if there isn't a gap before audio starts when starting stopping etc. However there's also a short noise using the Paris senderella, and in both cases I am wondering if some of the clicks/pops I've heard during playback occasionally are from the plugin.

For my usage, being 100% sure there's no jitter or other imperfections being introduced is critical. I'm not sure how that can be tested, but would be interested to know if this is a realistic expectation.

In any case, this is a fantastic project that opens up a lot of possibilities. By the way, I am also going to give VSTforx with the "remote sneder" a try (not free, but very cheap and there's a demo version). I may well use some of the other features of that plug anyway. Will report back on my experiences with all the above.