DSP - Tasks #978
finalize initial SpatLib implementation

2011-10-18 03:28 pm - Nils Peters

Status:	Closed	**Start date:**	2011-10-18
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	CNMAT 2012 workshop	**Spent time:**	0.00 hour
Branch:			

Description

Trond wrote:

_I've basically thought that the SpatLib DSP extension would accept m inlets, and mix them to n outlets. It would have a simple API for adding support for SpatDIF descriptors to a spat unit, and the message space would be extendable so that parameters specific to one spatialisation method could be implemented. And finally the mixing method might draw on all other DSP units available, including gaining, matrixing, delays, filtering, etc.

An initial AudioUnit wrapper for SpatLib would wrap it using multichannel input and output signals. Possibly the multichannel signals would be extended to be able to carry meta-information on what kind of signal we have (e.g. 1st order B-format, stereo, 2 or more mono signals, etc).

The one major issue that I haven't had a solution to earlier has been how the AudioUnit wrapper would deal with the parameter space changing depending on what unit one is using. It might be that the upcoming Max 6 added support for dictionaries might make this a whole lot easier. Anyway we already have dictionaries implemented in Jamoma Graph.

_

Related issues:

Related to DSP - Tasks # 153: TTMixDown class is missing	**Closed**	**2009-05-28**

History

#1 - 2011-10-26 07:15 am - Nils Peters

- Subject changed from finalize initial SpatLib implmentation to finalize initial SpatLib implementation

#2 - 2012-11-05 08:28 pm - Trond Lossius

- Target version changed from Kansas City Workshop Sprint 2011 to CNMAT 2012 workshop

#3 - 2013-02-08 06:58 pm - Trond Lossius

- Status changed from New to Closed

Closing this issue, as we have done further development of SpatLib recently.