MAX TYPE CONVERSION TABLE

TO

FROM

	bang	int	float	symbol	list	signal	matrix
bang	**button** **delay** **trigger** with the argument “l”	**counter**, **random**, **table**, **timer**, re-output current: **int**, **number** (number box), arithmetic objects, **value**/**pv**, etc.	re-output current: **flonum** (float number box), **float**, arithmetic objects, **value**/**pv**, etc.	re-output current: **message** (message box), **sprintf**, etc.	**coll** re-output current: **zl.reg**, **pack**, etc.	**click~** **count~**	**jit.noise** **jit.movie** **jit.desktop**
int	**button** **uzi**	**pipe** **table** many arithmetic objects **trigger** with the argument “l”	Objects expecting a float will accept an int and convert, **scale**	**tosymbol** **trigger** with the argument “s” **sprintf**, **coll**	**pack**/**pak** **zl.group** **message** (message box) **thresh**	**sig~** **selector~** **playlist~**	**jit.movie** “frame” message, **jit.playlist** **jit.matrixset** **switch**
float	**button**	Objects expecting an int will accept a float and truncate it to an int.	**pipe** **mtof**/**ftom** **atodb**/**dbtoa** many arithmetic objects **trigger** with the argument “l”	**tosymbol** **trigger** with the argument “s” **sprintf**	**pack**/**pak** **zl.group** **message** (message box) **thresh**	oscillators (**cycle~**, **rect~**, **saw~**, **phasor~**, etc.), **sig~** **adsr~**	**jit.matrix** “setall” message
symbol	**button**	**spell** **dict** **coll** **fromsymbol**	**dict** **coll** **fromsymbol**	**sprintf** **dict** **coll** **trigger** with the argument “l”	**pack / pak** **atoi** **zl.group** **dict** **coll** **fromsymbol**	“set” msg. to **index~**, **groove~**, **play~**, **waveform~**, etc.	“name” message to **jit.matrix**
list	**button**	**unpack** **iter** **zl.nth** / **zl.mth** **zl.len**	**unpack** **zl.nth** / **zl.mth** **zl.sum**	**unpack** **tosymbol** **sprintf** **zl.nth** / **zl.mth**	**vexpr** **zl.iter** **zl.reverse** **zl.join** **trigger** with the argument “l”	**line~** **curve~**	**jit.fill**
signal	**edge~** onset detectors such as **fzero~**	(Low-resolution signal analysis,e.g., audio to MIDI note number)	**snapshot~** **spike~** pitch estimators such as **fzero~** and **fiddle~**	(Shazam and Soundhound do this sort of thing: does Max?)	(Low-resolution signal analysis, e.g., audio to MIDI notes) Also see **capture~**	Most of MSP: filters, delay lines, etc., etc.	**jit.poke~** (See Tutorial 27 “Using MSP Audio in a Jitter Matrix”)
matrix	**button**	**jit.matrixinfo** “planecount” output	**jit.iter**	**jit.fpsgui** can tell you the name of a matrix	**jit.spill** **jit.iter** **jit.matrixinfo** “dim” output	**jit.peek**, (also see **spigot~**)	Most of Jitter: operators, effects, etc., etc.