Appendix C
Class Summary

The following is a quick reference to HMSL classes.

OB.MORPH

Superclass: OB.ELMNTS

Description: The most basic HMSL object, from which others are derived.

OB.SHAPE

Superclass: OB.MORPH

Description: Representation of ordered points in n-dimensional space. Contains the values in each dimension for each point.

OB.ENVELOPE

Superclass: OB.SHAPE

Description: Contains the points that define an envelope contour for an Amiga instrument.

What it does: Nothing. You can start and stop one, but not execute it.

OB.WAVEFORM

Superclass: OB.SHAPE

Description: Contains the points defining a waveform for an Amiga instrument.

OB.SAMPLE

Superclass: OB.WAVEFORM

Description: Defines the points for a sample to be played by an Amiga instrument.

OB.COLLECTION

Superclass: OB.MORPH

Description: Contains other executable morphs, such as other collections, players, jobs, structures, and productions. Also contains a number representing a nodal weight. Has an associated behavior which specifies how its components will be executed.

What it does: When executed will invoke each of its components in one of three ways, sequentially, in parallel, or using a custom behavior. The weight may optionally be used by a collection's behavior.

OB.PRODUCTION

Superclass: OB.COLLECTION

Description: Schedulable general-purpose word. Contains executable Forth word(s).

What it does: When executed, executes each of its component words in turn and terminates.

OB.JOB

Superclass: OB.PRODUCTION

Description: Tasked, user-definable set of functions, which has duration associated with it.

What it does: When executed, does its functions at intervals specified by its duration until it terminates itself. May call instruments, and alter its own parameters (especially its duration).

OB.ACTION

Superclass: OB.PRODUCTION

Description: Stimulus-response mechanism. Contains word that recognizes stimulus, word that executes response, and words to do something when the action is turned on or off from the Action Table.

What it does: When the action is turned on, executes an initialization word. While the action is scanned, invokes the stimulus word to look for a stimulus, and the response word to respond if necessary. When the action is turned off, executes a terminate word.

OB.STRUCTURE

Superclass: OB.COLLECTION

Description: Structure with an added grid of tendencies. A tendency is the probability of moving from one component morph to another component morph. This grid may be used by the structure's behavior.

What it does: As with an OB.COLLECTION, except that the behavior may utilize the tendencies.

OB.PLAYER

Superclass: OB.JOB

Description: Object that associates a shape with time. Contains a shape to be played, a duty cycle used for durational calculations, a number indicating which of the shape's dimensions is to be used for duration, and an instrument to be used to play the shape.

What it does: When executed opens associated instrument and uses it to play the shape. When done, closes the instrument.

OB.ACTION-TABLE

Superclass: OB.COLLECTION

Description: Contains actions, organized into four priorities. Also contains state variables indicating that the table is on or off, and behaviors telling the table how to scan itself.

What it does: When turned on (executed), executes its component actions in the order determined by the behavior that is selected. Note that only those actions that are themselves on will be executed. Terminates when it's turned off.

OB.TRANSLATOR

Superclass: OB.ARRAY

Description: Object for one-to-one mapping of input numbers to output numbers. Contains a table of such mappings, along with an offset to be added to the value and a modulus.

What it does: When invoked accepts a number and, using it as an index into the table and combined with the offset and modulus produces a translated number.

OB.TUNING

Superclass: OB.TRANSLATOR

Description: Translator to generate the notes corresponding to the scale degrees of a user-defined scale.

What it does: Behaves like a translator.

OB.TUNING.RATIOS

Superclass: OB.ELMNTS

Description: Object that stores a scale as a set of ratio pairs, used to generate period values for Amiga instruments from a base period (stored as 1/1).

What it does: When passed a scale degree maps it using the corresponding ratio and base period to a period value.

OB.INSTRUMENT

Superclass: OB.LIST

Description: Maps the values coming from a shape to a real instrument. Contains ON and OFF words along with functions for initializing and terminating the real instrument.

What it does: When opened from a player, initializes the real instrument. When invoked from a player translates the values coming from the shape to something the real instrument can use. When the player closes the instrument, sends any terminating stuff to the real instrument.

OB.MIDI.INSTRUMENT

Superclass: OB.INSTRUMENT

Description: Simple implementation of instrument interface to standard MIDI devices. Contains appropriate ON and OFF words, etc.

What it does: As above in the MIDI environment.

OB.AMIGA.INSTRUMENT

Superclass: OB.INSTRUMENT

Description: Implementation of instrument interface to the Amiga internal sound-generating hardware. Contains the same sorts of things as instruments.

What it does: As above for the Amiga. Allows additional support of Amiga specialties such as sampled waveforms, frequency and amplitude modulation.

OB.ALLOCATOR

Superclass: OB.BARRAY

Description: Mechanism for allocating resources (such as MIDI channels). Contains table of resources and their state (allocated/not allocated).

What it does: When allocated, returns the next resource in the table that isn't allocated already, and marks it as taken. When deallocated, marks that resource as free again. Used to automatically assign MIDI channels to MIDI instruments.