

Using Synth Matrices

for

Igor Engraver

(Document under Revision)

By Ray Fonk

Table of Contents

1. INTRODUCTION.....	2
2. LOADING SYNTH MATRICES.....	2
3. FORMAT OF A SYNTH MATRIX FILE.....	3
4. SYNTH MATRIX DETAILS	3
4.1 STRUCTURE OF A SYNTH MATRIX.....	4
4.2 SYNTHESIZER INFO.....	5
4.3 INSTRUMENT DEFINITIONS	8
4.3.1 <i>Instrument Classes</i>	8
4.3.2 <i>Instruments</i>	9
4.4 PLAYBACK DEFINITIONS.....	9
4.4.1 <i>Playback Attributes</i>	10
4.4.2 <i>Patch Modifiers</i>	12
5. EXAMPLES.....	13
APPENDIX A: INSTRUMENT CLASS HIERARCHY.....	15
APPENDIX B: INSTRUMENT SERIAL NUMBERS.....	17
APPENDIX C: SYNTH MATRIX KEYWORDS.....	21

Using Synth Matrices for Igor Engraver¹

1. Introduction

This document describes the “Synth Matrix” files that Igor Engraver 1.6 uses for MIDI playback.

Synth Matrices are the basis for Igor Engraver playback: no matrix, no playback. The synth matrices assign MIDI banks and patches for each synthesizer to instruments used in Igor, and so allow the user very detailed control over playback. If desired, this assignment can be different depending on musical context (dynamic, legato/staccato, playing style etc). It is also possible to keep different matrices for one synthesizer e.g. with specialised balances and patch settings for use with different styles of music. For instance, one might want to use one matrix when working with a Jazz combo, and another one when writing for a symphony orchestra.

Within Igor Engraver, you can allocate a particular instrument group to a particular synth. Igor will try to use that synthesizer for playback of that group, by efficient use of the available resources: you can have e.g. a 24 instrument setup playing back over the 16 MIDI channels that plays back most, if not all, instruments and notes, since usually not all instruments are used simultaneously. Only in case there are no more available MIDI channels on a synth at any time during playback will Igor decide to move the sounds to another synth, if available.

2. Loading Synth Matrices

Synth matrices are user readable text files, and reside in the Igor “Synths” folder. Each file contains a detailed description of a synthesizer, its sounds and parameters. This allows Igor to tailor its MIDI output to make the best possible use of the capabilities of each individual synthesizer.

When Igor encounters a synthesizer in the active MIDI setup (or the active OMS Studio configuration on the MAC), it looks for a corresponding synth matrix document in its "Synths" folder. It does this by scanning all the matrices it finds in the folder and loading the first matrix whose internal *model name* (see below) matches that of the synth in question. Note that the filename of the text file that makes up the synth matrix is not important, as

¹ This document was written by Ray Fonk based on “The Building Blocks of a Synth Matrix” by António Nunes and information provided on the Igor-Talk forum. The author wants to thank António Nunes for reviewing this document. The document is under revision, and edited by NoteHeads AB.

long as it ends on “.txt”. For example, if the MIDI setup contains two synthesizers named, respectively "QuickTime Music" and "Roland MOC-1", Igor will search in its "Synths" folder for two “.txt” files, whose internal model names are respectively "QuickTime Music" and "MOC-1" and load them. If you have multiple files with the same internal model name, Igor will only load the first file it finds. Therefore, if you want to use alternative matrices for the same synthesizer you must keep them in a separate folder and copy the relevant file to the Igor “Synths” folder.

If you make changes to a synth matrix that Igor is using, you can use the new settings in Igor without exiting. To do this, save the changed matrix file, but do not rename the file. Then in Igor Engraver choose the menu item:

Edit→MIDI→Re-read synth matrices

and the new settings will be active immediately. You do not need to restart Igor.

3. Format of a Synth Matrix file

The remainder of this document describes the format, structure and syntax of the Synth Matrix files. This allows you to modify existing matrices or create new ones.

Synth matrix files are in ASCII text format. Filenames can be anything as long as they end on the extension “.txt”.

The format of the files is straightforward: each entry in the matrix consists of a sequence of keywords (sometimes followed by a numeric value) in a certain order. The syntax which defines this order is described in the remainder of this document. Keywords start with a letter and are not case sensitive.

Keywords (and values) are separated from each other by “white space”, i.e. one or more spaces, newlines, tabs or combinations thereof. This means that matrix entries can be written on a single line or split over multiple lines as desired.

All text following a semicolon (“;”) until the end of a line is ignored. This can be used to add documentation and comments to a synth matrix document.

4. Synth Matrix Details

This section contains a detailed description of synth matrix entries. The following conventions are used in this document: all text appearing literally in the synth matrix text is printed in Courier typeface, e.g. `drum-channel`. A place-holder is indicated by angular brackets and needs to be replaced by an appropriate item as follows:

<string>	Indicates a string of characters enclosed in double quotes (“), e.g. “General MIDI”
<int>	Indicates an integer number, e.g. 1, +3, -4 etc.
<fp>	Indicates a floating point number, e.g. 0.5, +3.7, -1.05 etc.
<keyword> etc.	All other expressions in <...> and their replacements are explained in the text near where they appear.

Optional items are enclosed in square brackets []. Certain keywords have been capitalized to improve readability (note that synth matrix syntax is not case-sensitive).

4.1 Structure of a Synth Matrix

A Synth Matrix file basically consists of two parts:

1. Synthesizer Info
2. Playback definitions

The Synthesizer Info part contains a number of global settings including which describe the model name of the synthesizer for this matrix, author name and comments and general MIDI settings. Also included are global settings for volume and MIDI velocities and at least one MIDI bank definition.

The second part of the matrix contains a list of playback definitions for specific instruments or instrument classes. For each instrument in the piece that is being played on the current synth, Igor looks up the corresponding entry in the matrix.

Such an entry could look as follows:

```
; Entry for Flute
instrument 10200 attributes ORD patch GM 73
```

The entry starts with the unique serial number that identifies the instrument. The `attributes` keyword is followed by one or more keywords which define the musical context for which the current playback definition applies. The `ORD` attribute indicates the default context (“ordinario”). As will be seen below, one can specify a number of other attributes to define specific musical context. The `ORD` attribute used for playback of all other notes, i.e. all notes to which none of the specific attributes apply.

The playback is defined by the `patch` keyword, followed by a MIDI bank name, a MIDI patch number, and optionally by playback modifiers. In this case the `GM` (general MIDI) bank and patch `GM 73` (flute) is specified. More details of how to specify instruments, attributes, patches and modifiers are given below.

It is possible to have more than one playback definition for an instrument. For instance one could define:

```
; Flute
instrument 10200
  attributes ORD patch GM 73
  attributes PIANO patch GM 73
  attributes FORTE patch GM 73 volume 0.85
  attributes FAST patch GM 73 delay +15
  attributes FAST PIANO patch GM 73 delay +15
  attributes FAST FORTE patch GM 73 volume 0.85 delay +15
```

See the section on playback definitions for details of all keywords. This example assumes a GM bank has been defined.

It is not necessary to have entries for each instrument (although one can): each instrument belongs to a so-called instrument class, which in turn belongs to another instrument class (the parent class). If there is no entry for a particular instrument, Igor will instead look for its instrument class. The most general instrument class, to which all other classes belong is called INSTRUMENT. If there is no entry for the instrument class, Igor will use the entry for its parent class, etc. So one could have instead of the above:

```
instrument-class FLUTE attributes ORD patch GM 73
```

which would then apply for all instruments in the FLUTE class. The complete instrument hierarchy is given in Appendix A, and which instruments belong to which class is described in Appendix B.

At least one playback definition needs to be present in the matrix. Therefore the most basic synth matrix would have the following one entry for the general class INSTRUMENT, which makes everything playback with GM patch 0 (which sounds like a piano on most synths):

```
instrument-class INSTRUMENT attributes ORD patch GM 0
```

We now look in detail at the synthesizer info, the definitions of instrument classes and instruments and how to specify playback definitions for an instrument or instrument class.

4.2 Synthesizer info

The synth matrix starts with general information on the matrix file and the synthesizer it describes. The following keywords are used for this:

Manufacturer <string>	Specifies the name of manufacturer of the synthesizer. Make sure this name is correct: the model name in the matrix is a combination of the manufacturer name and the model number, and has to match exactly with the synth name in the OMS Studio Setup configuration.
-----------------------	--

Model <string>	Specifies the model number of the synthesizer. Make sure this too is correct: the model name in the matrix is a combination of the manufacturer name and the model number, and has to match exactly with the synth name in the OMS Studio Setup configuration.
Author <string>	Specifies the name of author of this matrix
Comment <string>	Allows specification of a short description of this synth matrix

Next follow some global MIDI settings for this synthesizer that apply to all the channels. The following keywords are used for this:

GM-standard	Indicates the synthesizer for this matrix conforms to GM standard
Non-GM-standard	Indicates the synthesizer for this matrix does not conform to GM standard
Drum-channel <int>	Specifies which MIDI channel is used as drum channel. This channel is referred to in playback definitions by <code>patch -1</code>
Patch-change-delay <int>	Specifies the time delay needed to send a MIDI patch-change command (in milliseconds).
Pitch-bend-range <int>	
Bank <name> <cmd-string>	Defines a new MIDI bank name and the corresponding bank change command.. See text for more details.
Velocities <int0>...<int9>	Defines the global MIDI velocity setting to be used for each dynamic level in Igor (from <i>pppp</i> through <i>ffff</i>). See text for more details.
Volumes <int0>...<int9>	Defines the global MIDI volume setting to be used for each dynamic level in Igor (from <i>pppp</i> through <i>ffff</i>). See text for more details.

Most of these are self explanatory. Here is some more information on how to use MIDI velocities and volumes to determine overall playback dynamics, and how to define MIDI banks.

Overall dynamic balance is controlled by the global `velocities` and `volumes` keywords which tell Igor how to translate dynamics into MIDI velocities and volumes. This would look as follows:

;;;	(-	-	PIANO	-	-)	(-	-	-	ORD	-	-	-)	(-	-	FORTE	-	-)
;;;	pppp	ppp	pp	p	mp	mf	f	ff	fff	ffff													
Velocities	14	20	27	37	47	60	85	105	115	120													
Volumes	14	20	27	37	47	60	85	105	115	120													

Each keyword is followed by ten values which are the global (default) values for MIDI velocities and volumes associated with each of the dynamic levels: *pppp* - *ppp* - *pp* - *p* - *mp* - *mf* - *f* - *ff* - *fff* - *ffff*. These global settings can be tailored to the synthesizer associated with this matrix. (Note that anything louder than *ffff* is played *ffff*, and anything softer than *pppp* is played *pppp*, which is the way musicians interpret it). Adjustments for volume can be made in the entries for individual instruments and classes.

The bank command defines a bank name and is followed by a string which defines the MIDI bank change command. (Note: the following requires some knowledge of the MIDI byte code format). The string contains (in hexadecimal byte code) the MIDI bank change command byte codes and timing values for the intervals between the sending of the MIDI command codes, if needed. The command byte codes are the MIDI control change (CC) commands which are synthesizer dependent. For most synthesizers the bank change command consist of a CC0 code, optionally followed by a CC32 if two bytes are needed to indicate the bank. The MIDI command bytes for a CC code are B0 through BF (for channels 0 – 15) and are followed by the controller number. For example, if on a Roland JV-1080, which uses two-byte bank definitions, you would want bank named "GM" to refer to bank 51-03, you would define:

```
Bank GM "B0+ch 00 51 10ms B0+ch 20 03 10ms C0+ch nn"
```

Here, B0+ch 00 is the CC0 command on the current channel (indicated by ch) and 51 is the first byte of the synthesizer dependent code for the desired bank. It is followed by a delay of 10ms, and a CC32 command (B0+ch 20) on the channel with 03 as the second byte of the bank code. Another 10 ms delay is followed by a program change code on this channel which changes the patch to patch nn (the actual patch number to be used will follow the bank name in the playback definition).

Each matrix should contain at least one bank definition, even if multiple banks are not supported. The most basic bank definition therefore just contains the MIDI program change command to change the patch, as follows:

```
Bank GM "C0+ch nn"
```

In summary, an example of a synthesizer info part in a matrix could look like this:

```
; Model name of the synthesizer as it appears in the MIDI set
; This is used by Igor to locate the appropriate matrix
Manufacturer "Creative"
Model "SB Live! A"

; Author's name and matrix/synthesizer description
;
Author "A.N.Other"
Comment "Description of the synth here"

; Some general MIDI info
```

```

;
GM-standard           ; conforms to GM standard
Drum-channel 10      ; drum channel #
Patch-change-delay 10 ; in milliseconds

; Bank definitions for Soud Blaster Live card
; Note: the SB uses only one byte for bank definitions
Bank GM      "B0+ch 00 00 10ms C0+ch nn" ; GM      => bank 0
Bank User1   "B0+ch 00 04 10ms C0+ch nn" ; User1   => bank 4

```

4.3 Instrument Definitions

Instruments in Igor are ordered in a strict hierarchy according to their properties (woodwind, brass, unpitched percussion, and so forth), called the **Instrument Class Hierarchy**. Each instrument also has a serial number, that is used to uniquely identify it in synth matrices. In a synth matrix there can be definitions for both instrument classes and individual instruments.

4.3.1 Instrument Classes

Instrument classes are used as defaults that Igor can use when a particular synthesizer does not have a sound patch for a specific instrument used in a piece. The root of the hierarchy is the instrument class "INSTRUMENT". All other instrument classes and instruments are a subcategory of this class.

The instrument class INSTRUMENT is defined in the matrix as follows:

```

;;; -----
;;; INSTRUMENT class
;;; -----
instrument-class INSTRUMENT attributes ORD patch GM 0

```

The first three comment lines (starting with “;”) have no effect. The keyword "instrument-class" is followed by the name of the class (in this case INSTRUMENT) and by a batch of playback definitions, that tell Igor what patches to use when playing back instruments that use this particular class designation.

A definition for the woodwind family of instruments could look like this:

```

instrument-class WOODWIND attributes ORD patch GM 13

```

The WOODWIND class ‘inherits’ from, or ‘belongs to’, the WIND class, which in turn inherits from the INSTRUMENT class. The effect of this is, that if the WOODWIND class is not defined in a matrix document, Igor will use the patch number defined by the WIND class, if present, or else the patch number defined by the INSTRUMENT class. A properly setup synth matrix document will therefore allow Igor to always choose the closest possible match to an instrument.

The structure of the complete instrument class hierarchy used in Igor can be found in Appendix A.

4.3.2 Instruments

Within each instrument class, playback can be defined for specific instruments. All instruments belong to a specific instrument class according to their properties (woodwind, brass, unpitched percussion, and so forth). Each instrument also has a serial number, which is used to uniquely identify it in synth matrices.

As an example, this is how the definition of a violin section could look:

```
;;; -----  
;;; VIOLIN class  
;;; -----  
instrument-class VIOLIN attributes ORD patch GM 48  
  
;Solo Violin  
instrument 60000 attributes ORD patch GM 40  
  
;Violin  
instrument 60050 attributes ORD patch GM 40  
; 1st Violins Section  
instrument 60100 attributes ORD patch GM 48
```

The instrument class definitions are there to allow for a default playback. If a piece uses, say, a 2nd violins section, but there is no definition for that in the matrix, the VIOLIN instrument class definition patch number will be used. As you can see, patch numbers can be used repeatedly throughout a matrix document. In this case the VIOLIN instrument class, and both the Violin and Solo Violin instruments use the same patch number for playback. The instrument 1st Violins Section uses another patch number.

An overview of all the instrument serial numbers, and the instrument classes they belong to can be found in Appendix B.

4.4 Playback Definitions

All musical (sounding) attributes can be controlled very precisely through use of the playback definitions in the synth matrix. One or more of these definition follow the name of the instrument class or instrument. Each definition looks as follows:

```
attributes <playback-attributes>  
    patch <bankname> <patchnumber> [<patch-modifiers>]
```

Playback attributes consist of one or more of the attribute keywords defined in section 4.4.1 below. The default playback attribute is ORD.

The `patch` keyword is followed by the bank name and the patch number which indicates the MIDI bank/patch combination to be used for playback. The patch number is an integer between 0 and 127. Bank names can be defined by the user in the synth info section of the matrix.

For playback on the MIDI drum channel, which is commonly used for unpitched percussion instruments, the syntax is somewhat different. In this case the reserved patch number `-1` is specified followed by the `KEY` keyword and a MIDI key number to specify the desired drum sound:

```
attributes <playback-attributes>
    patch -1 KEY <keynumber> [<patch-modifiers>]
```

If desired, the patch number can be followed by one or more patch-modifiers. Each modifier consists of a keyword, possibly followed by a value. See below for a description of possible modifier keywords and values.

4.4.1 Playback Attributes

Here is a list of keywords that can be used to specify the attributes of the music context for which you want to define playback behaviour:

ORD	Defines the normal, default playback behaviour ("ordinario").
FAST	Define playback behaviour for short note values. The transition point between ORD and FAST seems dependent on the inertia of instrument used in Igor.
SLOW	Define playback behaviour for long note values. The transition point between SLOW and ORD seems dependent on the inertia of instrument used in Igor.
FORTE	Defines playback behaviour when dynamic is <i>ff</i> or louder. Note that anything louder than <i>ffff</i> will be played as <i>ffff</i> .
PIANO	Defines playback behaviour when dynamic is <i>pp</i> or softer. Note that anything softer than <i>pppp</i> will be played as <i>pppp</i> .
FIRST-LOUDER	Defines playback behaviour for e.g. <i>fp</i> , <i>sfz</i> or similar expressions and is used for the initial louder part, which then cross fades to whatever comes after (usually ORD).
ATTACK-ONLY	Defines playback behaviour for very short note values (instrument dependent) Is used for patches which contain only very short samples of the initial attack, or for short marcatissimo string sounds.
SLUR	Defines playback for notes which are slurred to the next note.
LEGATO	Defines playback for notes which are played legato (e.g. slurred to the next note. Observe that the last note under a slur is not included)

STACCATO	Defines playback for notes which have a staccato dot attached
STACCATISSIMO	Defines playback for notes which have a staccatissimo wedge attached
TENUTO	Defines playback for notes which have a tenuto dash attached
TENUTO-STACCATO	Defines playback for notes which have a tenuto-staccato dash-dot attached
VIBRATO	Defines playback behaviour for “vibrato” playing (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
MOLTOVIB	Defines playback behaviour for “molto vibrato” playing (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
NOVIB	Defines playback behaviour for “non vibrato” playing in strings (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
PONTICELLO	Defines playback behaviour for “sul ponticello” playing in strings (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
TASTO	Defines playback behaviour for “sul tasto” playing in strings (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
PIZZICATO	Defines playback behaviour for “pizzicato” playing in strings (triggered by an entry text, or by a “+” note attachment)
BARTOK-PIZZ	Defines playback behaviour for “Bartók pizzicato” in strings (indicated by attachment)
COL-LEGNO	Defines playback behaviour for “col legno” playing in strings (indicated by entry text) <i>[This functionality will be implemented in a later version]</i>
TREM	Defines playback behaviour for tremolos
HARMONICS	
ARPEGGIO	Defines playback behaviour when the instrument plays “arpeggio” (defined by entry text)
NO-ARPEGGIO	Defines playback behaviour when the instrument plays “non arpeggio” (defined by entry text)
FLZ	Defines playback behaviour for Flatterzunge <i>[This functionality will be implemented in a later version]</i>
STRAIGHT MUTE	Subclass of MUTE. (MUTE can be regarded as "Straight Mute").
PEDAL	

MALLET	
SOFT-MALLET	
HARD-MALLET	
FELT-MALLET	
METAL-MALLET	
RUBBER-MALLET	
RIMSHOT	
STOPPED	Defines playback behaviour for stopped notes (on French Horns)
MUTE	Defines playback behaviour when the instrument plays “con sordino” or with mute (triggered by an entry text). Can be regarded as STRAIGHT MUTE.
CUP MUTE	Defines playback behaviour for a cup mute
HARMON MUTE	Defines playback behaviour for a Harmon mute
STIMME	
SOLO	
HAUPTSTIMME	Used with the special markings invented by Schönberg and can be used together with DYNAMIC and VOLUME modifiers to create solo effects.
CHORAL-STIMME	Used with the special markings invented by Schönberg and can be used together with DYNAMIC and VOLUME modifiers to create solo effects.
NEBEN-STIMME	Used with the special markings invented by Schönberg and can be used together with DYNAMIC and VOLUME modifiers to create solo effects.
UNTERSTIMME	Used with the special markings invented by Schönberg and can be used together with DYNAMIC and VOLUME modifiers to create solo effects.
HAUPTRHYTMUS	Used with the special markings invented by Schönberg and can be used together with DYNAMIC and VOLUME modifiers to create solo effects.

4.4.2 Patch Modifiers

The following keywords can be used after the patch specification to modify playback behaviour:

KEY <int>	Playback this MIDI key on current channel. Used with drum channels, which have a different (pitchless) sound associated with each key. (Reminder: MIDI key #60 corresponds to middle C).
DELAY <int>	Measured in milliseconds. Used to make e.g. a slow string sound start later. Note DELAY can also be used to make a sound start earlier. Just use a negative delay value.
RELEASE <int>	RELEASE affects the release time in a similar fashion as DELAY does the start time
OCTAVE <int>	Transposes the patch up or down by the specified number of octaves (1, -1, 2, -2, etc).
SEMITONES <int>	Transposes the patch up or down by the specified number of semitones (1, -1, 2, -2, etc)
DYNAMIC <int> DYNAMIC <fp>	Affects the MIDI velocity with which the notes are played. Can either be integer, in which case value is added to the original value, or floating-point, in which case it acts as a multiplier.
VOLUME <int> VOLUME <fp>	Affects the general volume of the channel used. Can either be integer, in which case value is added to the original value, or floating-point number, in which it is a multiplier. NOTE: Handle with care –messing around with channel volumes will use up more channels.
WEIGHT	
VELOCITY-SENSITIVE	
VELOCITY-INSENSITIVE	
VELOCITY-PERFORMING	
VELOCITY-NON-PERFORMING	

5. Examples

This section contains some illustrations of matrix entries taken from the GM matrix.

The simplest definition of an instrument would be:

```
; Violincello
instrument 62000 attributes ORD patch GM 42
```

This will make the violoncello always play back with General MIDI patch number 42. The ORD keyword defines the normal, default playback behaviour ("ordinario"). Let's say we want to change the dynamic performance of a cello. In the General MIDI matrix it is defined as follows:

```
; Violoncello
instrument 62000 attributes ORD patch GM 42 VOLUME 1.05
                attributes FORTE patch GM 42
                attributes PIANO patch GM 42 VOLUME 1.05
                attributes TREM patch GM 44 VOLUME 1.05
                attributes PIZZICATO patch GM 45 VOLUME 1.05
```

The volume of the violoncello is adjusted by +5% when the instrument plays anything softer than *ff*, and the patches for tremolo and pizzicato playback are defined.

Another way of using the VOLUME keyword, is to add an amount to the global MIDI value for each dynamic. In this case an integer value needs to be specified. As an example: when Igor Engraver has to play a fortissimo note for the violoncello it looks up the global value for *ff* (which is e.g. 105), and if it encounters the following definition in the matrix:

```
instrument 62000 attributes FORTE patch GM 42 VOLUME -10
```

it subtracts 10 units, so the sound will be played with MIDI velocity 95.

Keywords can be combined to specify particular situations as in the following example, again from the GM matrix:

```
; Flute
instrument 10200
    attributes ORD patch GM 73
    attributes PIANO patch GM 73
    attributes FORTE patch GM 73 volume 0.85
    attributes FAST patch GM 73 delay +15
    attributes FAST PIANO patch GM 73 delay +15
    attributes FAST FORTE patch GM 73 delay +15
    attributes ATTACK-ONLY patch GM 73 delay +15
    attributes ATTACK-ONLY PIANO patch GM 73 delay +15
    attributes ATTACK-ONLY FORTE patch GM 73 delay +15
```

The double bass entry in the GM matrix looks as follows:

```
instrument 63000 attributes ORD patch GM 43 OCTAVE -1
```

Here the OCTAVE modifier keyword makes the double bass sound one octave lower than notated.

As discussed, pitchless percussion instruments are handled through patch -1 (corresponding to the standard MIDI drum channel, which is specified in the top section of the synth matrix and is usually set to channel 10) and you select the sound you want by indicating a MIDI key through the KEY keyword. For example, the bass drum entry in the GM matrix is:

```
instrument 35000 attributes ORD patch -1 KEY 35
```

Appendix A: Instrument Class Hierarchy

The full instrument class hierarchy recognised by Igor is as follows (subclasses are indented):

```
INSTRUMENT
  WIND
    WOODWIND
      FLUTE
      RECORDER
      SINGLE-REED
        CLARINET
        HIGH-CLARINET
        MID-CLARINET
        LOW CLARINET
        BASS-CLARINET
        CONTRABASS-CLARINET
      SAXOPHONE
      DOUBLE-REED
        OBOE
        BASSOON
    BRASS
      FRENCH-HORN
      TRUMPET
        HIGH-TRUMPET
        MID-TRUMPET
        LOW-TRUMPET
      CORNET
      TROMBONE
      TUBA
        WAGNER-TUBA
  PERCUSSION
    PITCHED-PERCUSSION
      PITCHED-SKIN-PERCUSSION
      PITCHED-METAL-PERCUSSION
        VIBRAPHONE
        STEEL-DRUM
      PITCHED-WOOD-PERCUSSION
        XYLOPHONE
        MARIMBA
      PITCHED-PERCUSSION-EFFECT
    UNPITCHED-PERCUSSION
      UNPITCHED-SKIN-PERCUSSION
        SNARE-DRUM
      UNPITCHED-METAL-PERCUSSION
        CYMBAL
      UNPITCHED-WOOD-PERCUSSION
      RATTLE-PERCUSSION
      UNPITCHED-PERCUSSION-EFFECT
  PLUCKED-STRING
    ACOUSTIC-PLUCKED-STRING
      NYLON-ACOUSTIC-PLUCKED-STRING
      HARP
      METAL-ACOUSTIC-PLUCKED-STRING
      ELECTRIC-PLUCKED-STRING
        ELECTRIC-GUITAR
        ELECTRIC-BASS
  KEYBOARD
    WIND-KEYBOARD
      ORGAN
        ELECTRIC-ORGAN
    REED-KEYBOARD
    MALLET-KEYBOARD
      PIANO
        ELECTRIC-PIANO
    HISTORIC-KEYBOARD
  VOCAL
    HIGH-VOCAL
    LOW-VOCAL
    CHORUS
```

CHILDREN-CHORUS
FEMALE-CHORUS
MALE-CHORUS
MIXED-CHORUS
BOWED-STRING
HIGH-BOWED-STRING
VIOLIN
VIOLA
LOW-BOWED-STRING
VIOLONCELLO
DOUBLEBASS

Appendix B: Instrument Serial Numbers

Here is a list of all the unique instrument serial numbers used in Igor, and the instrument classes they belong to:

Piccolo	10100	flute
Flute in E flat	10150	flute
Flute	10200	flute
Alto Flute	10300	flute
Bass Flute	10400	flute
Sopranino Recorder	10500	recorder
Soprano Recorder	10600	recorder
Alto Recorder	10700	recorder
Tenor Recorder	10800	recorder
Bass Recorder	10900	recorder
Oboe in E flat	10990	oboe
Oboe	11000	oboe
Oboe d'amore	11100	oboe
English Horn	11200	oboe
Oboe da caccia	11300	oboe
Baritone Oboe	11350	oboe
Heckelphone	11400	oboe
Clarinet in E flat	12000	high-clarinet
Clarinet in D	12100	high-clarinet
Clarinet in C	12200	mid-clarinet
Clarinet in B flat	12300	mid-clarinet
Clarinet in A	12400	mid-clarinet
Alto Clarinet in E flat	12450	bass-clarinet
Basset Horn	12500	bass-clarinet
Basset Horn (Strauss)	12550	bass-clarinet
Bass Clarinet (French)	12600	bass-clarinet
Bass Clarinet (German)	12700	bass-clarinet
Contrabass Clarinet (French)	12800	contrabass-clarinet
Contrabass Clarinet (German)	12900	contrabass-clarinet
Sopranino Saxophone in E flat	13000	saxophone
Soprano Saxophone in B flat	13100	saxophone
Alto Saxophone in E flat	13200	saxophone
Tenor Saxophone in B flat	13300	saxophone
Baritone Saxophone in E flat	13400	saxophone
Bass Saxophone in B flat	13450	saxophone
Contrabass Saxophone in E flat	13500	saxophone
Bassoon	14000	bassoon
Contrabassoon	14100	bassoon
French Horn in C alto	20000	french-horn
French Horn in B flat alto	20050	french-horn
French Horn in A	20100	french-horn
French Horn in A flat	20150	french-horn
French Horn in G	20200	french-horn
French Horn in F	20250	french-horn
French Horn in E	20300	french-horn
French Horn in E flat	20350	french-horn
French Horn in D	20400	french-horn
French Horn in C basso	20450	french-horn
French Horn in B flat basso	20500	french-horn
Alto Horn in F	20600	french-horn
Alto Horn in E flat	20650	french-horn
Mellophonium in F	20700	french-horn
Mellophonium in E flat	20750	french-horn
Mellophone in F	20800	french-horn
Mellophone in E flat	20850	french-horn
Trumpet in F	21000	high-trumpet
Trumpet in E flat	21100	high-trumpet
Trumpet in D	21200	high-trumpet
Trumpet in C	21300	mid-trumpet
Trumpet in B flat	21400	mid-trumpet
Bass Trumpet in E flat	21500	low-trumpet
Flugelhorn	21600	low-trumpet
Cornet in B flat	21700	cornet
Cornet in A	21800	cornet

Soprano Trombone	22000	trombone
Alto Trombone	22100	trombone
Tenor Trombone	22200	trombone
Trombone	22300	trombone
Bass Trombone	22400	trombone
Contrabass Trombone	22500	trombone
Wagner Tuba in B flat	23000	wagner-tuba
Wagner Tuba in F	23100	wagner-tuba
Euphonium	23200	tuba
Baritone	23300	tuba
Tuba	23400	tuba
Bass Tuba	23500	tuba
Contrabass Tuba	23600	tuba
Kettledrums	30000	pitched-skin-percussion
Glockenspiel	31000	pitched-metal-percussion
Tubular Bells	31100	pitched-metal-percussion
Gongs	31200	pitched-metal-percussion
Antique Cymbals	31300	pitched-metal-percussion
Vibraphone (single-staff)	31400	vibraphone
Vibraphone (two-staff)	31500	vibraphone
Soprano Steel Drum	31600	steel-drum
Alto Steel Drum	31610	steel-drum
Guitar Steel Drum	31620	steel-drum
Tenor Steel Drum	31630	steel-drum
Cello Steel Drum	31640	steel-drum
Bass Steel Drum	31650	steel-drum
Almglocken	31700	pitched-metal-percussion
Musical Saw	31750	pitched-metal-percussion
Tubaphone	31800	pitched-metal-percussion
Cimbalon	31850	pitched-metal-percussion
Dulcimer	31900	pitched-metal-percussion
Zither	31950	pitched-metal-percussion
Xylophone (single-staff)	32000	xylophone
Xylophone (two-staff)	32100	xylophone
Marimba (single-staff)	32200	marimba
Marimba (double-staff)	32300	marimba
Xylorimba	32400	marimba
Keyed Harmonica	32500	pitched-percussion-effect
Whistling	32600	vocal
Sleigh Bells	33000	rattle-percussion
Tambourine	33100	rattle-percussion
Snare Drum	34000	snare-drum
Piccolo Snare Drum	34100	snare-drum
Military Drum	34200	snare-drum
Bass Drum	35000	unpitched-skin-percussion
Tom-tom	35100	unpitched-skin-percussion
Bongo drums	35200	unpitched-skin-percussion
Timbales	35300	unpitched-skin-percussion
Chinese tom-tom	35400	unpitched-skin-percussion
Tabla	35500	unpitched-skin-percussion
Kick Drum	35600	unpitched-skin-percussion
Triangle	36000	unpitched-skin-percussion
Suspended Cymbal	36100	cymbal
Cymbals	36110	cymbal
Tam-Tam	36120	cymbal
Finger cymbals	36130	cymbal
Sizzle cymbal	36140	cymbal
Hi-hat cymbal	36150	cymbal
Chinese cymbal	36160	cymbal
Crash cymbal	36170	cymbal
Ride cymbal	36180	cymbal
Flexatone	36200	pitched-metal-percussion
Cowbells	36300	unpitched-metal-percussion
Bell plate	36400	unpitched-metal-percussion
Automobile brake drums	36500	unpitched-metal-percussion
Anvil	36600	unpitched-metal-percussion
Tibetan Bowl	36700	unpitched-metal-percussion
Claves	37000	unpitched-wood-percussion
Wood Block	37100	unpitched-wood-percussion
Temple Block	37200	unpitched-wood-percussion
Castanets	37300	unpitched-wood-percussion
Whip	37400	unpitched-wood-percussion
Slit drum	37500	unpitched-wood-percussion
Drum sticks	37550	unpitched-wood-percussion
Log Drum	37600	unpitched-wood-percussion

Wind Chimes	38000	rattle-percussion
Bamboo Chimes	38100	rattle-percussion
Rattle	38200	rattle-percussion
Maracas	38300	rattle-percussion
Güiro	38400	rattle-percussion
Cabasa	38500	rattle-percussion
Quijada	38600	rattle-percussion
Vibraslap	38700	rattle-percussion
Washboard	38800	rattle-percussion
Wind Machine	39000	unpitched-percussion-effect
Chains	39050	unpitched-percussion-effect
Police whistle	39100	unpitched-percussion-effect
Lion's roar	39150	unpitched-percussion-effect
Bird whistle	39200	unpitched-percussion-effect
Duck call	39250	unpitched-percussion-effect
Siren	39300	unpitched-percussion-effect
Auto horn	39350	unpitched-percussion-effect
Thundersheet	39400	unpitched-percussion-effect
Sandpaper blocks	39450	unpitched-percussion-effect
Klaxon horn	39500	unpitched-percussion-effect
Pistol shot	39550	unpitched-percussion-effect
Handclap	39600	unpitched-percussion-effect
Jew's harp	39700	unpitched-percussion-effect
Auto horn	39750	unpitched-percussion-effect
Typewriter	39800	unpitched-percussion-effect
Finger snaps	39850	unpitched-percussion-effect
Harp	40000	harp
Guitar	40100	nylon-acoustic-plucked-string
Irish Harp	40110	nylon-acoustic-plucked-string
Shamisen	40120	nylon-acoustic-plucked-string
Alto guitar	40130	nylon-acoustic-plucked-string
Koto	40140	nylon-acoustic-plucked-string
Lute	40150	nylon-acoustic-plucked-string
Teorb	40160	nylon-acoustic-plucked-string
Archiliuto	40170	nylon-acoustic-plucked-string
Mandolin	40200	metal-acoustic-plucked-string
Steel-string guitar	40300	metal-acoustic-plucked-string
Ukulele	40400	metal-acoustic-plucked-string
Banjo	40500	metal-acoustic-plucked-string
12-string guitar	40510	metal-acoustic-plucked-string
Bouzouki	40520	metal-acoustic-plucked-string
Sitar	40530	metal-acoustic-plucked-string
Electric Guitar	40600	electric-guitar
Jazz Guitar	40650	electric-guitar
Electric Bass	40700	electric-bass
Fretless Bass	40750	electric-bass
Pedal Steel Guitar	40800	electric-plucked-string
Piano	41000	piano
Grand Piano	41100	piano
Electric Piano	41110	electric-piano
Honky-Tonk Piano	41120	piano
Upright Piano	41130	piano
Toy Piano	41140	piano
Celesta	41200	mallet-keyboard
Harpsichord	41300	historic-keyboard
Ondes Martenot	42000	electric-organ
Electric Organ	42050	electric-organ
Moog Bass	42060	electric-organ
Clavinet	42070	electric-organ
Synthesizer	42080	electric-organ
Cottage Organ	42100	reed-keyboard
Accordion	42150	reed-keyboard
Musette	42160	reed-keyboard
Bandoneon	42170	reed-keyboard
Regal	42175	reed-keyboard
Keyboard	42180	keyboard
Organ (without pedals)	42200	organ
Organ (with pedals)	42300	organ
Shakuhachi	43000	flute
Panpipe	43050	flute
Harmonica	43100	single-reed
Ocarina	43150	flute
Cornett	43200	single-reed
Bagpipe	43250	double-reed
Dulcian	43300	double-reed

Rankett	43350	double-reed
Crumhorn	43400	double-reed
Ney	43450	double-reed
Hü ch'in	45000	bowed-string
Viola da gamba	45100	bowed-string
Rebec	45200	bowed-string
Fiddle	45300	bowed-string
Hurdy-Gurdy	45400	bowed-string
Violone	45500	bowed-string
Vielle	45600	bowed-string
Crwth	45700	bowed-string
Kantele	45750	pitched-metal-percussion
Viola d'amore	45800	bowed-string
Traverse Flute	45850	flute
Tape	47000	instrument
Boys' Chorus	50000	children-chorus
Chorus SA	50100	female-chorus
Chorus SSAA	50200	female-chorus
Chorus TB	50300	male-chorus
Chorus TTBB	50400	male-chorus
Chorus SAB	50500	mixed-chorus
Chorus SATB	50600	mixed-chorus
Chorus SSAATTBB	50700	mixed-chorus
Chorus SMzATBarB	50800	mixed-chorus
Soprano	51000	high-vocal
Mezzo-Soprano	51100	high-vocal
Alto	51200	high-vocal
Counter-tenor	51300	high-vocal
Tenor	51400	low-vocal
Baritone	51500	low-vocal
Bass	51600	low-vocal
Vocal/Solo	51700	vocal
Solo Violin	60000	violin
Violin	60050	violin
Violin I	60100	violin
Violin I div in 2	60150	violin
Violin I div in 3	60200	violin
Violin I div in 4	60250	violin
Violin I div in 6	60260	violin
Violin II	60300	violin
Violin II div in 2	60350	violin
Violin II div in 3	60400	violin
Violin II div in 4	60450	violin
Violin II div in 6	60460	violin
Violin III	60500	violin
Viola	61000	viola
Solo Viola	61100	viola
Violas	61200	viola
Violas div in 2	61300	viola
Violas div in 3	61400	viola
Violas div in 4	61500	viola
Violas div in 6	61510	viola
Violoncello	62000	violoncello
Solo Violoncello	62100	violoncello
Violoncellos	62200	violoncello
Violoncellos div in 2	62300	violoncello
Violoncellos div in 3	62400	violoncello
Violoncellos div in 4	62500	violoncello
Violoncellos div in 6	62510	violoncello
Solo Doublebass	63000	doublebass
Doublebass	63100	doublebass
Doublebasses	63200	doublebass
Doublebasses div in 2	63300	doublebass
Doublebasses div in 3	63400	doublebass
Doublebasses div in 4	63500	doublebass

Appendix C: Synth Matrix Keywords

The following table presents in alphabetical order the keywords recognised in synth matrix syntax and described in this document. This list does not include instrument class names; see Appendix A for those.

A		HARMONICS	12
ARPEGGIO	12	HAUPTRHYTMUS	14
ATTACK-ONLY	11	HAUPTSTIMME	13
Attribute	10	I	
Author	4, 6	Instrument	8
B		instrument-class	5, 8
Bank	6, 7, 8	int	4
BARTOK-PIZZ	12	K	
C		KEY	10, 14
CHORAL-STIMME	13	keyword	4
COL-LEGNO	12	L	
Comment	3, 6	LEGATO	11
CUP MUTE	13	M	
D		MALLET	12
DELAY	14	Manufacturer	6
Drum-channel	3, 6, 8	METAL-MALLET	12
DYNAMIC	13, 14	Model	6
F		MOLTOVIB	11
FAST	5, 10	MUTE	12, 13
FELT-MALLET	12	N	
FIRST-LOUDER	11	NEBEN-STIMME	13
FLZ	12	NO-ARPEGGIO	12
FORTE	5, 10	Non-GM-standard	6
fp	4	NOVIB	11
G		O	
GM-standard	6, 8	OCTAVE	14, 16
H		ORD	5, 7, 10, 15
HARD-MALLET	12	P	
HARMON MUTE	13	Patch-change-delay	6

PEDAL	12	string	4
PIANO	5, 7, 10	T	
Pitch-bend-range	6	TASTO	11
PIZZICATO	11	TENUTO	11
PONTICELLO	11	TENUTO-STACCATO	11
R		TREM	12, 15
RELEASE	14	U	
RIMSHOT	13	UNTERSTIMME	13
RUBBER-MALLET	13	W	
S		WEIGHT	14
SEMITONES	14	Velocities	6, 7
SLOW	10	VELOCITY-INSENSITIVE	15
SLUR	11	VELOCITY-NON-PERFORMING	15
SOFT-MALLET	12	VELOCITY-PERFORMING	15
SOLO	13	VELOCITY-SENSITIVE	14
STACCATISSIMO	11	VIBRATO	11
STACCATO	11	VOLUME	13, 14, 15
STIMME	13	Volumes	6, 7
STOPPED	13		
STRAIGHT MUTE	12, 13		