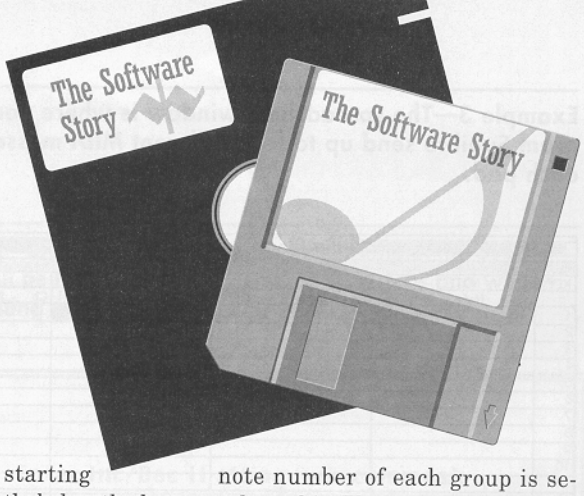
Sybil 3.0: Real-Time Performance Software for Macintosh and IBM Compatibles

By Norm Weinberg



SYBIL IS NOT A SEQUENCER, NOTATION PROGRAM, SYNTH editor or MIDI file player. Sybil is a real-time performance program unlike anything else on the market. The program is available for both Macintosh and IBM compatible computers; I tested only the Macintosh version. Sybil ran successfully on a Mac Plus, a Powerbook 140 and a PowerMac 7100.

The Sybil package comes with a master disk and two short manuals. The user's manual explains all the various features available with Sybil, and the shorter tutorial manual takes the user through several tutorial files that are included on the master disk. While the program is not copy protected, it does require that the serial number be entered the first time the program is launched. But every time I tried to enter my serial number, the program wouldn't accept it as being valid. A quick call to the technical people at ghs confirmed that there was a glitch with the validation process. They offered to send an unprotected copy of the program and it arrived within a few days. According to the company, the validation process will be removed, making installation a snap.

HOW IT WORKS

One of the impressive features of a computer is its number-crunching power. Since MIDI is essentially a musical language that involves numbers, computers can perform basic mathematical operations on MIDI data in real-time or in non-real-time. Many functions and features in software-based sequencers operate in non-real-time. For example, if you sequence a C Major triad and then transpose the sequence by a major third, the software simply "adds four" to each MIDI note number (note 60 becomes 64, etc.). The result of such a mathematical transposition would be an E Major triad.

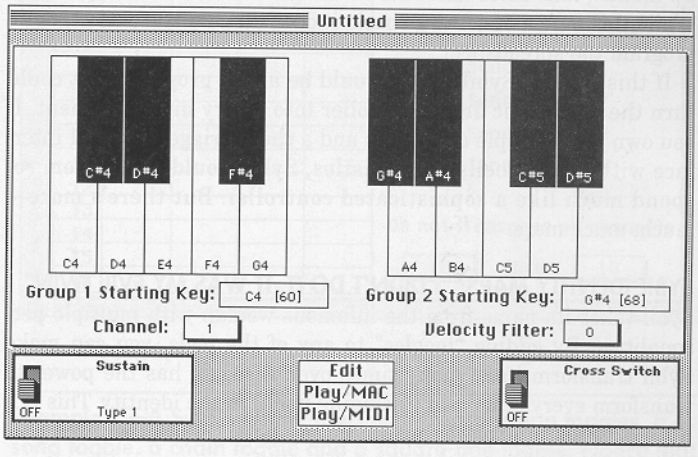
Instruments such as the KAT DrumKAT, the Roland Pad 80 and the Yamaha TMX system perform similar mathematical tricks in real-time. In such cases, the software inside the unit might be programmed to do something like this: When a signal is received on trigger input 6, output MIDI note number 36 on MIDI channel two, note 40 on channel two, and note 43 on channel three. The result of such a program would be the output of a C Major triad with the strike of a single pad—pretty cool for electronic drummers who use sticks on surfaces instead of fingers on keys!

But Sybil works in a slightly different manner. By inserting a computer running Sybil between your master controller (the MIDI instrument you're playing) and your sound generator(s), Sybil alters MIDI data in real-time. Believe it or not, there is no perceptible delay in the transformation process. Here's a simple example: You can program Sybil so that when it receives MIDI note number 36 over MIDI channel one, it will output up to four MIDI note numbers on up to four MIDI channels.

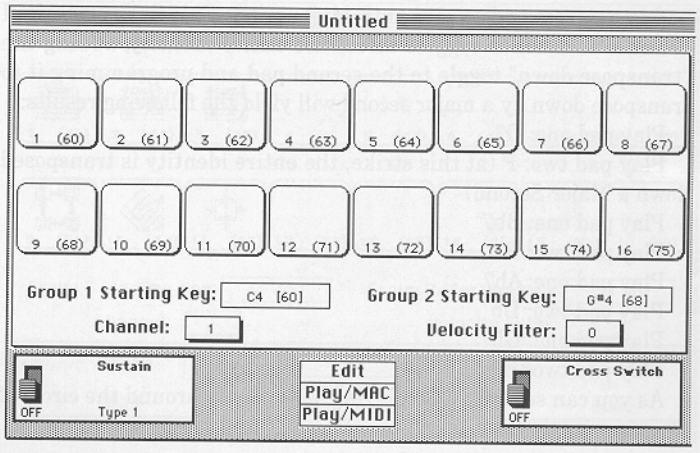
To get a better idea how Sybil operates, take a look at Examples 1 and 2. At first, you'll notice that you can view the Sybil "Identity" as either a piano keyboard or as a set of pads (or buttons). In the center of the screen you can see two groups of eight keys or pads offering a total of sixteen MIDI input note numbers. Each group can have a different starting note number, but the others in the group must be sequential. The starting note number of each group is selected directly below the key or pad graphic. There are pop-up selections for both the MIDI input channel and a velocity filter. If the velocity filter is set to a high level, Sybil will ignore all MIDI note numbers below the selected velocity. In addition, Sybil will ignore all information outside of the range of the sixteen pads or keys. This MIDI information will not be lost, it will simply pass through the program unaffected and go directly to the sound generator.

Notice (at the lower-center of the screen) that you can play Sybil from both a MIDI controller (Play/MIDI) and from the Macintosh keyboard itself (Play/MAC). When playing from the Macintosh keyboard, the numbers 1-8 activate the first set of eight pads and the letters q-i activate the second set of pads. As might be expected, when playing Sybil from the computer's keyboard, you can't perform with velocity, and all your attacks will be at the same volume. With a click on the Edit button, you're taken to the "Spreadsheet" window.

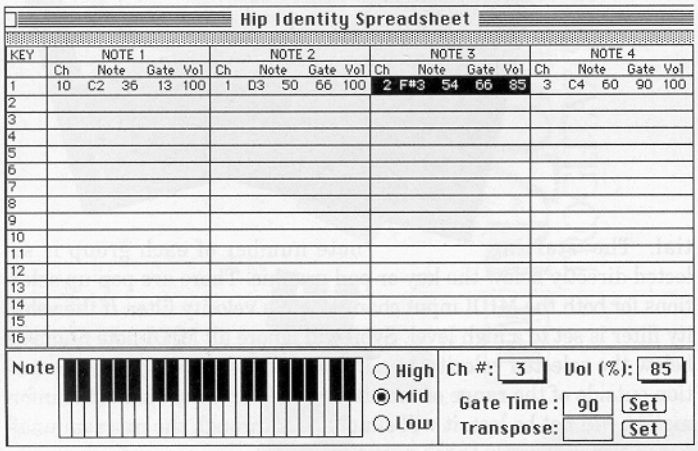
Example 1—Sybil's main screen shown as a piano keyboard.



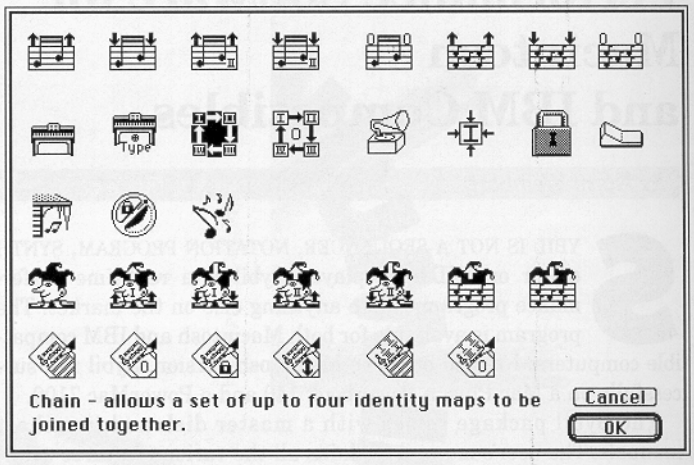
Example 2—Sybil's main screen shown as a set of pads or buttons.



Example 3—The spreadsheet window is where you program Sybil to send up to four different MIDI messages for each pad.



Example 4—Sybil "toggles." Notice that selecting a toggle brings up a brief description of its function. All toggles are defined in the text.



THE SPREADSHEET

In the spreadsheet window (Example 3), you program Sybil to output up to four notes for each input pad. Notice that you can program not only the MIDI note number and channel, but also the gate time (duration) and relative velocity.

In all honesty, I found that programming the spreadsheet was a little cumbersome. You can't simply type numbers directly in the boxes inside the window. Instead, you must click on a note in the keyboard, select MIDI channels and velocity percentage from pop-up menus, and enter amounts in the gate-time area individually. While this isn't a major drawback, it does take a little extra time to program the spreadsheet.

If this was all Sybil did, it would be a cool program that could turn the most basic drum controller into a very hip instrument. If you own just a couple of triggers and a cheap trigger-to-MIDI interface without any bells and whistles, Sybil would make them respond much like a sophisticated controller. But there's more—much, much more.

SYBIL IDENTITY MAPS: "I DIDN'T DO IT, IT WAS MY EVIL TWIN"

Sybil takes its name from the infamous woman with multiple personalities. By adding "toggles" to any of the pads, you can make Sybil transform itself many times over. A toggle has the power to transform every other pad (including itself) in the identity. This is a little difficult to explain, but I'll do my best. (Remember, this program is like nothing you've seen before.)

Okay, let's say you've programmed the first pad to play a C Dominant 7th chord (C, E, G, Bb) over MIDI channel one, and the second pad is programmed to play an F Major triad (F, A, C, F) over MIDI channel two. Since toggles affect the entire identity, adding the "transpose down" toggle to the second pad and programming it to transpose down by a major second will yield the following results:

Play pad one: C7

Play pad two: F (at this strike, the entire identity is transposed down a Major Second)

Play pad one: Bb7

Play pad two: Eb

Play pad one: Ab7

Play pad two: Db

Play pad one: Gb7

Play pad two: B

As you can see, you'll be playing V7-I chords around the circle of fourths—pretty neat, huh?!

Sybil includes a variety of interesting toggles as shown in Example 4. From left to right, they are:

Row One:

Increment One—transposes all outgoing notes up by selected number of half steps.

Decrement One—transposes all outgoing notes down by selected number of half steps.

Increment Two—transposes all outgoing notes up through a series of intervals. (See Example 5 for an example of an Increment Two series.)

Decrement Two—transposes all outgoing notes down through a series of intervals.

Reset Inc/Dec—resets transpositions of all outgoing notes back to zero.

Microtonal Increment—shifts pitches up by the given microtonal amount (performed by taking control of the pitch wheel).

Microtonal Decrement—shifts pitches down by the given microtonal amount.

Microtonal Reset—resets microtonally incremented pitches.

Row Two:

Sustain—alters the sustain characteristics of outgoing notes.

Sustain Type—toggles between sustain types one and two.

Chain—allows a set of up to four identity maps to be joined together.

Chain Reset—returns to the first link in the chain of identity maps.

Program Change—sends up to four sets of patch changes across channels one through sixteen (See Example 6).

Square One—returns all toggles to their default states.

Toggle Lock—locks all toggles in their current state until next time this toggle is hit.

Mute—turns Sybil off until next attack of mute toggle. Notes on the input channel are treated as MIDI thru.

Row Three:

Hang—sustains the last chord played until next hang toggle attack or repetition of same chord.

Channel Lock—locks/unlocks transposition on the MIDI channel range indicated in the Program Change window.

Harmonize—turns harmonies on or off.

Row Four:

Wild—assumes the value of and plays the last set of outgoing notes.

Wild Increment One—transposes the last set of outgoing notes

up by specified interval.

Wild Decrement One—transposes the last set of outgoing notes down by specified interval.

Wild Increment Two—transposes the last set of outgoing notes up through a series of intervals.

Wild Decrement Two—transposes the last set of outgoing notes down through a series of intervals.

Microtonal Wild Increment—transposes the last set of outgoing notes up by the specified microtonal interval.

Microtonal Wild Decrement—transposes the last set of outgoing notes down by the specified microtonal interval.

Row Five:

Song—a list of notes/chords that are played one at a time in order, per toggle hit.

Reset Song—Moves to the beginning of the song.

Song Offset Lock—locks/unlocks the movement through a song.

Song Inc/Dec Lock—locks/unlocks song column one from transposing all outgoing notes.

Song Position—moves to the location of the next song pointer.

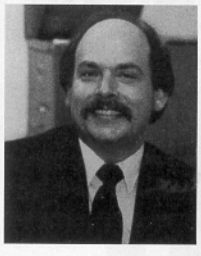
Song Position Reset—moves to the location of the first song pointer.

As you can see from the descriptions of the toggles, Sybil has the power to turn the most basic electronic percussion setup into the most sophisticated controller available (take a look at Example 7 for an illustration of a complete identity map). There isn't a controller on the market that offers as many features, as much flexibility and as many options as running Sybil. This is truly a unique and important program. With a few pads, a laptop computer and a sound generator, you can become a portable one-man band and play music that would be impossible without Sybil. With a little thought and planning, you can use Sybil to create music that has never been heard before—and control it all with your sticks!

For More Information: ghs Corporation, P.O. Box 136, Battle Creek MI 49016. Voice: (800) 388-4447; FAX: (800) 860-6913.

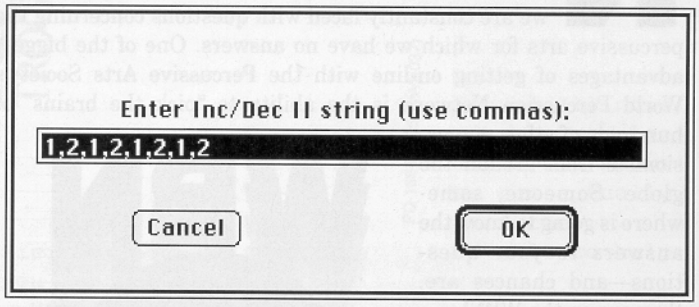
Retail Price: $139.00

Required Hardware: MIDI Controller, MIDI Interface, MIDI Sound Generator. **PN**

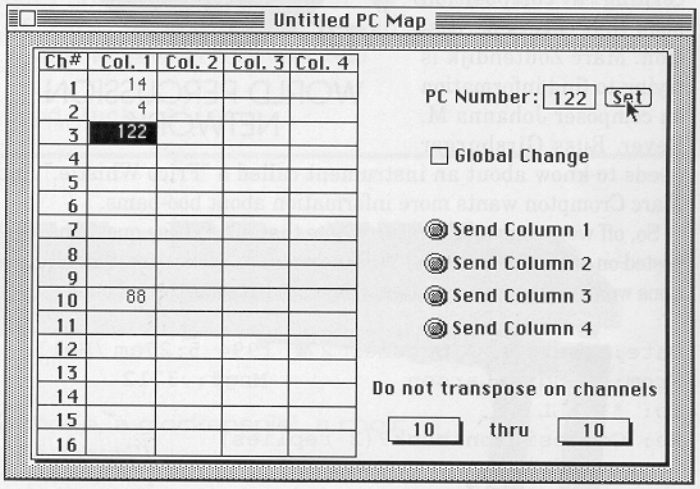


Dr. Norm Weinberg *is a Professor of Music at Del Mar College in Corpus Christi, Texas and Principal Timpanist/Percussionist with the Corpus Christi Symphony. He serves as Associate Editor of* Percussive Notes *and as Chairperson for the PAS World Percussion Network Committee.*

Example 5—Increment Two String. In this example, the Sybil identity will be transposed by a series of half-steps and whole-steps, creating an octatonic scale.



Example 6—The Patch Change Map allowing up to four sets of program change messages across all sixteen MIDI channels.



Example 7—A Sybil identity map with transposition toggles, a song toggle, a chain toggle and a square-one toggle. Notice that velocity cross switching will call up another identity ("Blues").