mimetics™

corporation

Dedication:

The staff at Mimetics would like to dedicate SoundScape to Scott W. Gibbs. As co-founder of Syntauri Corporation and designer/programmer of Metatrak, the first commercially available music sequencer, Scott put a new creativity tool in the hands of thousands of artists and was responsible for the birth of what has become a multi-million dollar industry. Scott founded Mimetics in early 1984 to further explore music software and begin a relationship with MIDI. Many of Scott's original concepts and ideas can be found in SoundScape, and with his guidance we bring you the "next generation" in music software, a "professional software MIDI music studio".

About t

A native Cali
amounts c f
master's deg
Stanford, he
ambition a
suggest that
music softwa
his attribu s
and noted tha
of free tim h

Acknowl

Mimetics w
Bruce Car
Boddicke
of FAUG c
Michael B
media sup c
ideas and ad
lines and to
for the "n t

Technical C

About the programmer:

A native Californian, **Todor Fay** grew up in Sierra Madre and can eat large amounts of food very fast. Todor wrote SoundScape while completing a master's degree in computer science at Stanford University. Previous to Stanford, he attended CalArts, where he studied film. Todor's childhood ambition was to become a MAD INVENTOR. Since any wise soul will suggest that you have to be crazy if you think you can make a living writing music software, Todor seems to have realised this dream. Todor lists among his attributes fallen arches and bad knees from hard days on the soccer field, and noted that the nicest thing about completing SoundScape is the abundance of free time he'll now have to brew beer and make music.

Acknowledgements:

Mimetics would like to extend thanks to **Chris Potter, Eric Myers,** and **Bruce Carroll** for their tireless dedication to debugging, **Michael Boddicker** and **Tom Scott** for their public support, **Paul Montgomery** of FAUG for his continued commitment to this project and the Amiga, **Michael Boom** for turning out a great manual, **Peggy Herrington** for media support, and **Sam Dicker** for RAMSample and countless nights of ideas and advice. Additional thanks to **Jim Weigel** for manning the front lines and to **Jay Miner** and the Amiga Whiz Kids for producing the machine for the "next generation".

Technical Coordinator: **Bob Hoover**

Table of Contents:

Chapter Ei
Echo M od
Trigge M
Match M
Transp se
Seeing M

Chapter Ii
The Playe
The MIDI
The M DI
The MIDI
The Cl k
Saving e
Additiona
IFF SM J

Introduction

Welcome to SoundScape Pro, the first of a series of programs using the SoundScape music operating system for the Amiga computer. SoundScape Pro (referred to as SoundScape throughout this manual) is a program that takes full advantage of all the facilities found in the SoundScape music operating system to record, play, and edit music. Although SoundScape Pro has the power and flexibility to satisfy the professional musician, it's also simple enough for beginners to use once they understand how the parts of SoundScape work. This manual is written to introduce you to SoundScape gently and easily, explaining important concepts and details to the novice user, and giving examples so beginning and advanced users can learn to use SoundScape to their best advantage.

Section One of the manual introduces you to SoundScape. The first chapter of the section shows you how to start running SoundScape and how to use the SoundScape controls. The second chapter gives you a quick tour of SoundScape, where you'll play a song and look at all the different windows available in SoundScape. The third chapter gives a brief explanation of MIDI (Musical Instrument Digital Interface) for those who haven't worked with it before, and explains how the different parts of SoundScape use MIDI to communicate with each other.

Section Two of the manual is a tutorial that teaches you how to use all the different parts of SoundScape. The first chapter of the section shows you

how to use the Console Keyboard and the Sampled Sound Instrument Player. The second chapter introduces you to the Tape Deck, and shows you how to play back a score. In the third chapter, you'll find information about recording your own scores, and in the fourth chapter that follows, more information about how to edit your scores. The fifth chapter of the section teaches you about advanced Tape Deck features, and the final chapter covers parts of SoundScape not covered in the other chapters: the MIDI Mixer, the Player Piano, and other features.

Although this manual is written to help beginners work with SoundScape, it assumes you know how to use the Amiga computer. If you're not familiar with the workings of the Amiga, you should first read *Introduction to Amiga*, a manual included with the Amiga computer. *Introduction to Amiga* shows you how to use the mouse, the Workbench, disk drives, and other Amiga hardware and software. You'll use the information you learn in *Introduction to Amiga* throughout this manual.

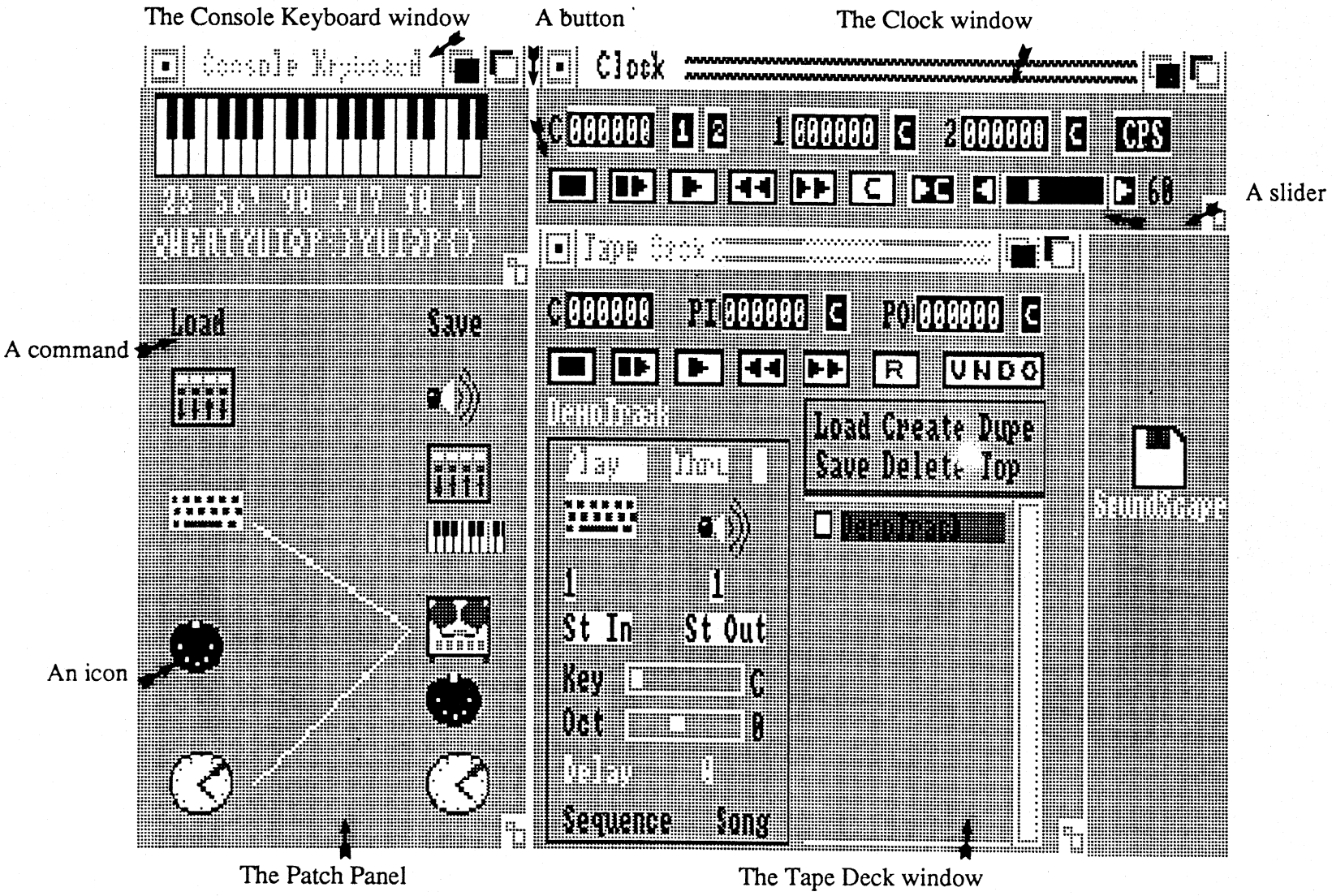
If you want to work with SoundScape on a very technical level (for example, writing programs in C that use sections of SoundScape), you can find detailed technical information in materials included with the SoundScape Developer's System. Contact Mimetics at the address on the SoundScape registration card for information on its availability.

One final note: the colors of the software and the Workbench are described in this manual the way they appear using standard SoundScape Workbench colors: grey, white, black, and red. If you have changed SoundScape's Workbench colors with the Preferences program, your colors won't match the colors described in this manual. If you get confused by the color descriptions, simply change the Workbench colors back to the original SoundScape colors.

If you now have your Amiga ready and your disk in hand, then you're ready to try out SoundScape. Read on....

§§§§§§§§§§§§§§§§

Figure 1: A full page, labelled screen dump of SoundScape.



Section One:

An Introduction to SoundScape

Figure 1: A full page, labelled screen dump of SoundScape.

Chapter One: Running SoundScape

SoundScape is simple to run. You can use it with just a basic Amiga system (the console with a single internal disk drive, a keyboard, a mouse, and a monitor), or with added external music hardware such as synthesizers and samplers. Once you have the hardware set up, you can start SoundScape using the Amiga's Workbench, and control SoundScape with simple controls in SoundScape's windows.

SETTING UP THE HARDWARE

If you have any external music hardware, such as a synthesizer, drum box, or any other device with a MIDI interface, connect it to the Amiga using a Mimetics SoundScape MIDI interface or other Amiga-compatible MIDI interface. The interface plugs into the Amiga's serial port. You then use MIDI cables (available where you bought your MIDI synthesizer or other MIDI equipment) to connect the MIDI device to the Amiga's MIDI interface. Follow the instructions included with the MIDI interface to make the connections.

SoundScape can use the Mimetics SoundScape Sampler, a piece of external hardware, to record sampled sounds (explained in a later chapter). The Sampler plugs into the Amiga's second controller port, and has jacks that you

can use to connect a microphone or other audio signal source to the Sampler. To make the connections, follow the instructions in the SoundScape Sampler manual. (If you're interested in buying either the SoundScape MIDI interface or the SoundScape Sampler and would like more information, write to Mimetics at the address on the registration card.)

If you want to feed the sound from the Amiga through a stereo or PA system for better sound than you can get from the small speaker in your video monitor, simply connect the two audio ports on the back of the Amiga's console to a pair of line level inputs in the back of a stereo amplifier using standard audio cables. (Line level inputs are usually labelled LINE IN or AUX IN or can be the inputs for a compact disk or tape deck.) You can also connect the Amiga audio ports directly to the line level inputs of a tape recorder so you can record music directly from your Amiga.

RUNNING THE SOFTWARE

SoundScape comes on a disk that doesn't include the Amiga Workbench so that you can install SoundScape on new versions of Workbench as Commodore-Amiga releases them. To install SoundScape on the version of Workbench you use with your Amiga, (a procedure you only have to follow once), follow these steps:

1. Turn on the Amiga and the video monitor.

2. Insert the Kickstart disk in the internal disk drive and let it load.

3. When the Amiga asks for the Workbench disk, eject the Kickstart disk and insert your Amiga Workbench disk.

4. Make a duplicate of the Workbench disk using the Disk Copy directions in *Introduction to Amiga* .

5. Change the name of the duplicate disk from "Copy of Workbench" to "Workbench" using the Rename instructions in *Introduction to Amiga* .

6. Eject your new Workbench disk and write "SoundScape Workbench" on its label so you know it's the version you'll use to run SoundScape.

7. Reboot the Amiga by pressing the CTRL key and the two Amiga keys simultaneously.

8. When the Amiga asks for the Workbench disk, insert the SoundScape disk. A CLI screen will appear, with a "1>" prompt.

9. Type "Execute SSinstall" and then press the RETURN key.*

10. Follow the directions that appear on the screen to complete the installation (it will ask you to swap disks, using the Workbench copy you just made.)

11. When you're finished with the installation procedure, you can eject your disks and reboot the Amiga with the CTRL and Amiga keys.

*One thing to note: If your copy of Workbench has too many files on it, then you may not be able to install SoundScape on it for lack of disk space. If that's so, delete a few files like the Demo programs from your Workbench copy disk and try once again.

That's all there is to integrating SoundScape with Workbench. From now on, you'll use both the SoundScape Workbench and the SoundScape disks to run SoundScape. Once you've integrated SoundScape, you run SoundScape just like you run most other Amiga programs. Make sure you have your Amiga Kickstart and SoundScape Workbench disks on hand along with your SoundScape disk, then follow these steps:

1. Turn on the Amiga and the video monitor (and the monitor speaker or stereo system if you use an external audio system for sound instead of the video monitor's speaker) if they aren't already on.

2. Turn on any external MIDI devices you have connected to the Amiga.

3. Insert the Kickstart disk in the internal disk drive and let it load if you just turned on your Amiga. If you just rebooted your Amiga, skip this step.

4. When the Amiga asks for the Workbench disk, eject the Kickstart disk and insert the SoundScape Workbench disk to start running the Amiga Workbench.

5. When the Workbench is loaded, insert the SoundScape disk.

6. When the SoundScape disk icon appears, double-click on the icon to open up the SoundScape disk window.

7. Double-click the SoundScape program icon in the SoundScape disk window to run SoundScape.

Two windows appear on the Workbench: a tiny window titled "SoundScape", and a larger window titled "Patch Panel". SoundScape is now ready for you to use. (If you're impatient, you can skip to the next chapter, **A Tour of SoundScape**, to start using it without an introduction to it.)

WORKING WITH MODULES IN THE PATCH PANEL

SoundScape is not one program, but a collection of programs called *modules* that run simultaneously and independently of each other. The modules perform different musical activities, and some have their own windows on the Amiga's Workbench where you can control them. You can see some of

these module windows in Figure 1 at the beginning of this chapter. SoundScape modules use the underlying SoundScape music operating system to create sounds, interpret musical information, and pass musical information back and forth to each other using the MIDI standard, described in Chapter Three, **MIDI and SoundScape**. The modules on the SoundScape disk are:

- The **Console Keyboard**, which turns the Amiga's console keyboard into a "piano" keyboard; it reads keystrokes as notes instead of letters and numbers, and passes them on to other modules. It also reads some keystrokes as stop, start, and advance signals, and sends them as timing and control information to other modules.

- The **MIDI In** module, which reads signals from external MIDI devices (like synthesizers and drum boxes) connected to the Amiga with a MIDI adaptor, and passes the signals on to other modules.

- The **Clock** creates timing signals that it can send out to other modules, providing timing for the Tape Deck module and synchronizing other modules and external MIDI devices.

- The **Sampled Sound Player** accepts notes from the Console Keyboard module or other modules and plays them using sampled sound instruments stored in the Amiga's internal audio channels. It also lets you create your own sampled sound instruments using the SoundScape Sampler (hardware that plugs into the Amiga), and lets you modify and edit existing sampled sound instruments to create totally new sounding instruments.

- The **Tape Deck** records incoming signals from other modules and external MIDI devices, and lets you edit and mix those recordings. It plays back recordings, sending out signals to the other modules and

external MIDI devices. You can alter the speed and transposition of the Tape Deck's playback.

• The **Player Piano** turns incoming signals into a visual display, showing notes on the screen as colored keys playing on a piano keyboard.

• The **MIDI Out** module sends signals from SoundScape out to external MIDI devices attached to the Amiga with a MIDI adaptor.

• The **MIDI Mixer** accepts input from any module on any of 16 different MIDI channels (discussed in Chapter Three). It mixes together any signals that come in on the same channel, and sends the mixed signals out on the same channel.

• The **Patch Panel** lets you connect the different SoundScape modules together so they can exchange signals, and also lets you open the windows for different modules so you can work with their controls.

The Patch Panel, the window you see on the Workbench now titled "Patch Panel", has two columns of icons that represent different SoundScape modules. You can see the icons in Figure 1 at the beginning of this chapter. At the top of the two columns are the words "Load" and "Save". These words are commands, not icons, that you'll learn to use in later chapters.

The modules in the left column of the Patch Panel present an input signal to the Patch Panel (that is, music information comes into the Patch Panel from these modules). From top to bottom, the modules and the icons that represent them are:

1. The **MIDI Mixer**

2. The **Console Keyboard**

3. The **MIDI In module**

4. The **Clock**

The modules in the right column accept an output signal from the Patch Panel (that is, music information goes from the Patch Panel into these modules). From top to bottom, they are:

1. The **Sampled Sound Player**
2. The **MIDI Mixer**
3. The **Player Piano**
4. The **Tape Deck**
5. The **MIDI Out module**
6. The **Clock**

To connect an input module on the left to an output module on the right so the input module can send music information to the output module, first click on the input module, then click on the output module. A line appears between the modules to show they're connected. To disconnect two modules, repeat the same action. The line will disappear.

When you connect two modules, a window or two may open on the Workbench. These windows contain the controls for the modules you clicked on in the Patch Panel. Module windows will usually open if they aren't already open when you connect their icons in the Patch Panel. If a window doesn't automatically open, or you want to open a module window without connecting the module, you can double-click on the module's icon (by pointing to the icon with the mouse pointer and clicking the left mouse button twice quickly) to open the module's window. (The MIDI Mixer and the MIDI Out modules have no controls, so double-clicking on their icons has no effect.)

USING SOUNDSCAPE CONTROLS

When you look in the SoundScape module windows, you can see three kinds of controls (labelled in Figure 1 at the beginning of this chapter):

Buttons are symbols in the window that are sometimes enclosed by a box and sometimes not. You *click on* a button with the mouse pointer to start an action. (You click on an object by pointing to it with the mouse pointer and then clicking the left mouse button.)

Commands are words in the window. You *select* a command by clicking on it with the mouse pointer. Selecting a command usually opens a new window or starts a sequence of events.

Sliders are elongated hollow rectangles with a small white rectangle inside. You *set* a slider by dragging the small white rectangle within the outer rectangle with the mouse pointer. (You drag something by putting the mouse pointer on it, holding down the left mouse button, and then rolling the mouse.)

In addition to SoundScape controls, the module windows also use standard Workbench window gadgets around the window's frame--close gadgets, front gadgets, back gadgets, and sizing gadgets--that you can use to move the windows around the screen, change their size, and make them appear in front of or behind other windows. (See *Introduction to Amiga* for more information on these gadgets.)

RUNNING OTHER PROGRAMS WITH SOUNDSCAPE

SoundScape is a multi-tasking program; that is, you can run other programs on the Amiga while you run SoundScape. As an example, try running the Notepad program on the Workbench disk:

1. Click on the back gadgets of the SoundScape windows until the Workbench disk window appears. (If you haven't opened the Workbench disk window, move the SoundScape windows out of the upper right corner of the Workbench until you see the disk icon, then double-click on the icon.)

2. Double-click on the Utilities drawer to open the Utilities window (which includes the NotePad icon).

3. Double-click on the NotePad icon to start the Notepad program working. A window titled "Notepad" should appear on the Workbench.

4. Try typing in the Notepad.

5. At any time you want, you can click on the back gadget of the Notepad window and bring the SoundScape windows to the front so you can work with them. SoundScape and the Notepad program are running simultaneously.

6. When you're finished playing with Notepad, click on its close button to clear it from memory.

QUITTING SOUNDSCAPE

Of course, at this point, you haven't done anything with SoundScape and probably don't want to stop using it yet, but at some point in later examples, you may want to stop for a rest. To quit SoundScape at any time when you're

done using it, c
ask you if you'r
on the "Yes" b
memory. All th
their window .)
you to close it b

If you click th
SoundScape, th
and leave the r
window repr
to load Sound
down Sound c
from memory.
can close So
in the Workben
click in its clos

done using it, close the tiny window titled "SoundScape". SoundScape will ask you if you're sure you want to close SoundScape down completely. Click on the "Yes" button to stop SoundScape from running and clear it from memory. All the opened modules will be automatically closed (including their windows.) If you have an edit window open, SoundScape will wait for you to close it before shutting down.

If you click the "No" button when you're asked if you want to close SoundScape, then SoundScape will only close the small SoundScape window, and leave the rest of SoundScape running. This is because the SoundScape window represents a program running on Workbench whose sole purpose is to load SoundScape. When you close the window and choose not to shut down SoundScape, then only the little loader program stops and is cleared from memory. If you want to get the small SoundScape window back so you can close SoundScape completely, just double-click on the SoundScape icon in the Workbench. The SoundScape window will reappear, and you can then click in its close box once again.

§§§§§§§§§§§§§§§§

Chapter Two: A Tour of SoundScape

A quick tour of SoundScape will show you some of SoundScape's capabilities and help you get acquainted with its different parts. This chapter shows you a simple way to play music using SoundScape, and then shows you all the different windows that SoundScape uses on the Amiga's Workbench. The tour is simple and short; all you have to do is run SoundScape, prop this book up by your Amiga, and follow the instructions as you read.

PLAYING MUSIC

Playing music on SoundScape is easy: you just load an *environment* from the SoundScape disk and start it playing. An environment is a song stored on disk with instrument and control settings stored with it for each of the SoundScape modules. When you load an environment, SoundScape automatically makes all the necessary connections in the Patch Panel, opens the appropriate modules, and sets all the module settings the way they were saved in the environment.

One of the environments on the SoundScape disk is named "SimpleDemo", and is set up to use the Amiga's own sound synthesis to play music, and to use

the Amiga k
hear it:

1. Click on the
   Patch Pane
   Workbe h

Figure 2: h

2. Point to the
   click the fi
   SoundScape
   labelled " l
   already open

3. Try play g
   QWERTYU
   speaker. I
   SoundScape's

4. To play th s
   pointer to the
   bottom of he

the Amiga's keyboard to let you perform live music. Follow these steps to hear it:

1. Click on the "Load" command on the top of the left column of icons in the Patch Panel. A window labelled "Environments" will appear on the Workbench. This is the environment requester window, seen in Figure 2.

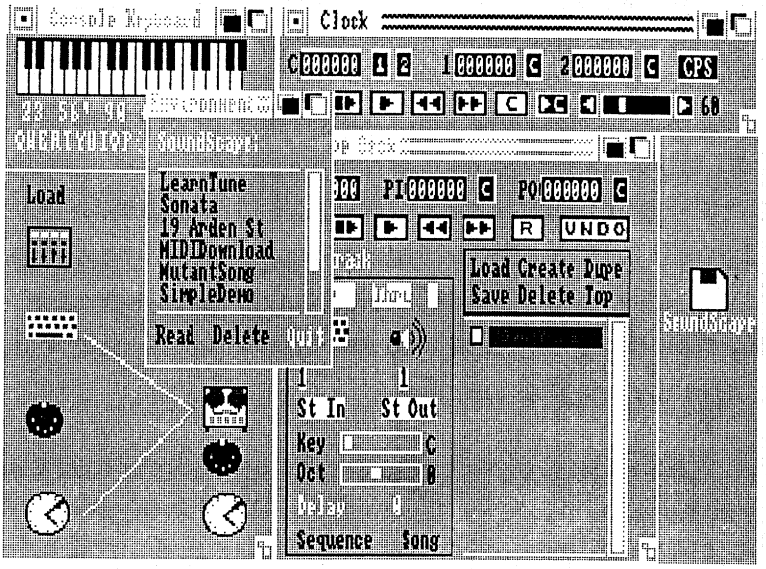


Figure 2: The environment requester window.

2. Point to the name "SimpleDemo" in the requester window and double-click the left mouse button. The environment will load, and as it loads, SoundScape makes connections in the Patch Panel and opens three windows labelled "Clock", "Tape Deck", and "Console Keyboard" (if they aren't already open).

3. Try playing the Amiga's keyboard by pressing any of these keys: QWERTYUIOP[]. You should hear notes coming from the monitor speaker. (If you don't, check the volume setting). This is one of SoundScape's sampled sound instruments.

4. To play the song that SoundScape loaded with the environment, move the pointer to the Clock window. There is a row of white rectangles along the bottom of the window (seen in Figure 1 at the beginning of Chapter One).

These are clock control buttons. Click on the second button from the left to start the song playing. (You can stop it at any point by clicking on the the button all the way to the left.)

5. You can change the tempo of the song's playback by dragging the slider in the lower right corner of the Clock window. Drag the small white rectangle in the middle of the slider right to speed it up, left to slow it down.

6. As the song plays back, you can change the instrument that plays the lead line: Close to the middle of the Tape Deck window is a box labelled "St Out". Just above it is the number "1". Click on this number. It will change to 2, and you'll hear a new instrument playing in the score. If you click again, it will change to 3, but nothing will play because no sound was loaded for channel 3. In all, there are 16 channels, so cycle through them to get back to the original sound.

7. When you want to stop playing the song, click on the leftmost button in the clock window.

You've just played (and played with) your first SoundScape song!

LOOKING AT WINDOWS

Playing songs and performing on the Amiga keyboard is just a small part of what you can do with SoundScape. To get an idea of SoundScape's other features, you can look at all the different windows available. Each window has its own set of activities and controls that run those activities.

THE CONSOLE KEYBOARD WINDOW

SoundScape's Console Keyboard module interprets the keys you press on the Amiga's keyboard as notes of music. It also interprets some keys as control

signals (you'll learn about them in the next section of this manual). You can open the Console Keyboard's window by connecting the Console Keyboard icon to another icon in the Patch Panel, or by double-clicking on the Console Keyboard icon. The Console Keyboard window shows the Console Keyboard module's activity. If you click on the window and then play notes on the Amiga's keyboard, you'll see keys flash in red on the keyboard in the window. These flashing notes show you what pitches you're playing on the Amiga's keyboard. (You can see the Console Keyboard window in Figure 1 at the beginning of Chapter One.)

THE EDIT CONSOLE KEYBOARD WINDOW

If you double-click on the Console Keyboard icon in the Patch Panel, a window titled "Edit Console Keyboard" (shown in Figure 3) will open with the Console Keyboard window. The sliders in this window control the way the Console Keyboard module sends its music information to other modules in the Patch Panel. Under the sliders are boxes where you can set numbers for the space bar and the function keys. These numbers pace the timing of Tape Deck recording when you use a special recording mode called "step time" (discussed in a later chapter in the next section). When you're done looking at the Edit Console Keyboard window, click on its close gadget to close it.

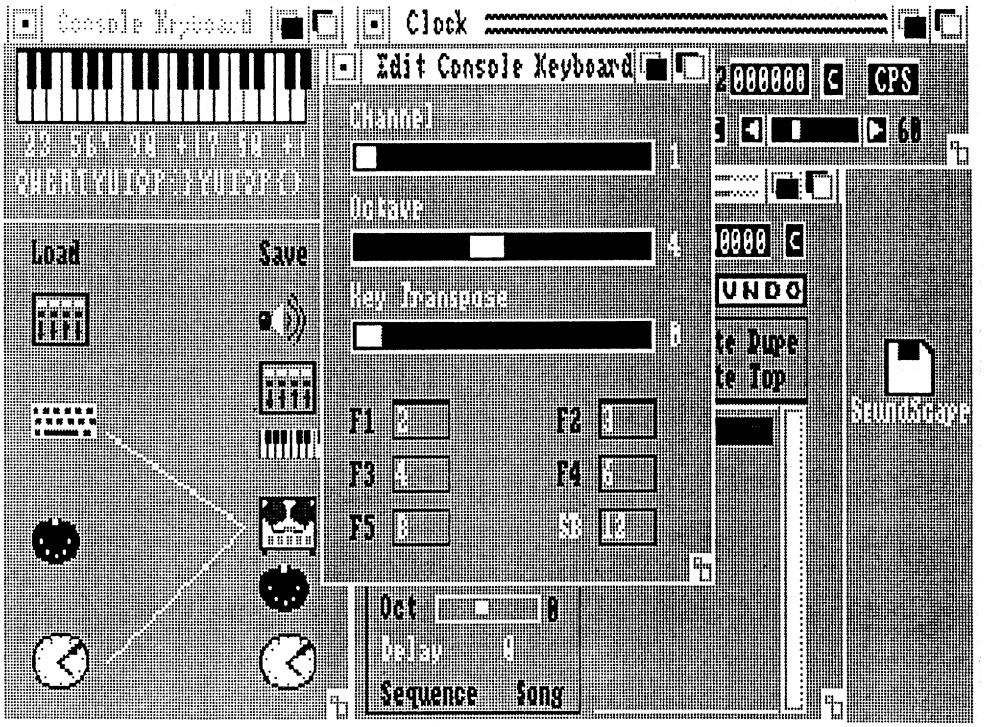


Figure 3: The Edit Console Keyboard window.

THE SET UP MIDI IN WINDOW

If you double-click on the MIDI In icon in the Patch Panel, a small window titled "Set Up Midi In" opens (shown in Figure 4). The window has a single slider that you can use to set the MIDI channel that an external synthesizer or other device uses to send music information to SoundScape. Close the window when you're done looking at it.

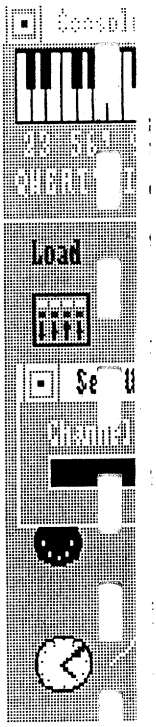


Figure 4:

THE SOU

When you p
SoundScap p
double-click
titled "Sou d
sampled so n
with. You ca
listed in lin s
environment,
commands t
instruments

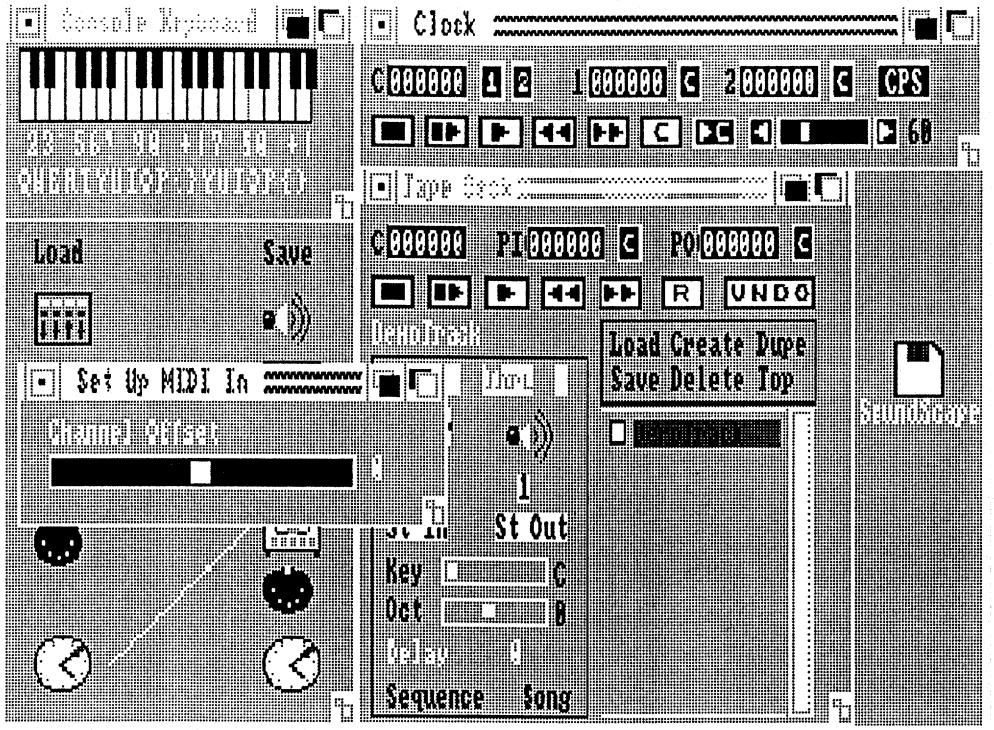


Figure 4: The Set Up MIDI In window.

THE SOUNDS WINDOW

When you played a song using the environment in the last example, SoundScape played the notes using the Sampled Sound Player module. If you double-click on the Sample Sound Player icon in the Patch Panel, a window titled "Sounds" will open (shown in Figure 5). This window lists all the sampled sound instruments the Sampled Sound Player can use to play notes with. You can see the two instruments used in the environment you played listed in lines 1 and 2. The 14 other lines have no instruments loaded in this environment, so they list "No Name". The "Load All" and "Save All" commands at the bottom of the window let you save and load whole banks of instruments to disk.

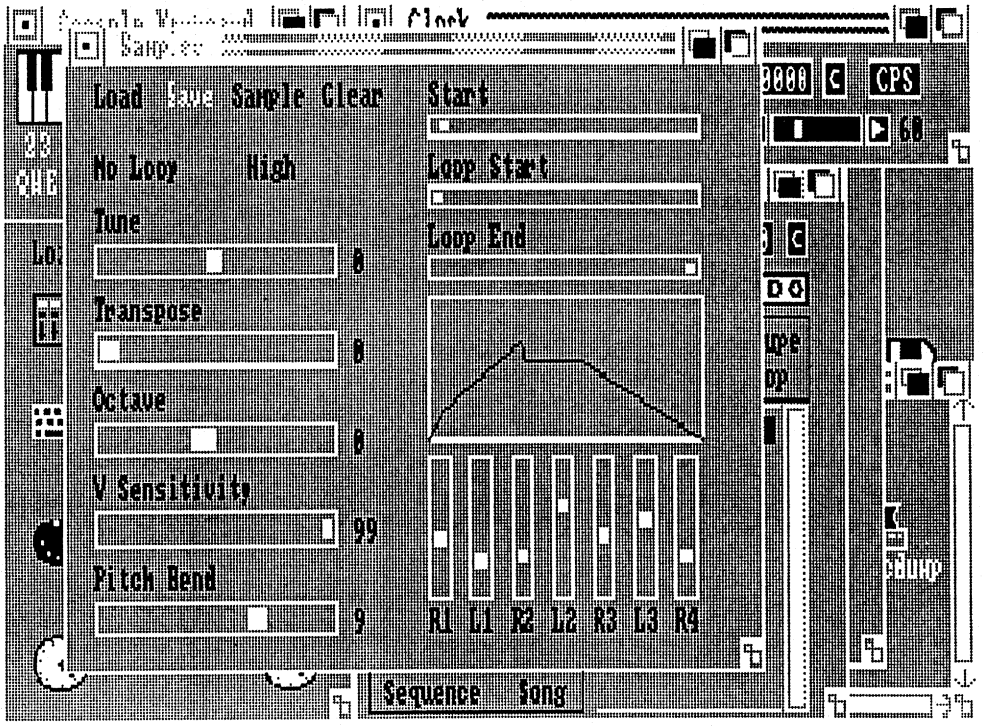


Figure 5: The Sounds and Sampler windows.

THE SAMPLER WINDOW

If you double-click on any of the instrument names or the No Names in the Sounds window, a window titled "Sampler" will open up (shown in Figure 5). The controls in this window set the characteristics of the sampled sound instrument you double-clicked on--the way it sounds, and the way it responds to incoming music information. The Load and Save commands here let you save instruments you create here to disk, and to load instruments previously saved from disk.

THE SAMPLE WINDOW

If you click on the Sample command in the top of the window, a window titled "Sample" will open (shown in Figure 6). The controls in this window let you work with a SoundScape Sampler attached to the Amiga to record your own sampled sound instruments from outside audio sources. When you're done looking at this window, close it, then close both the Sampler and Sounds windows to clear up the Workbench.

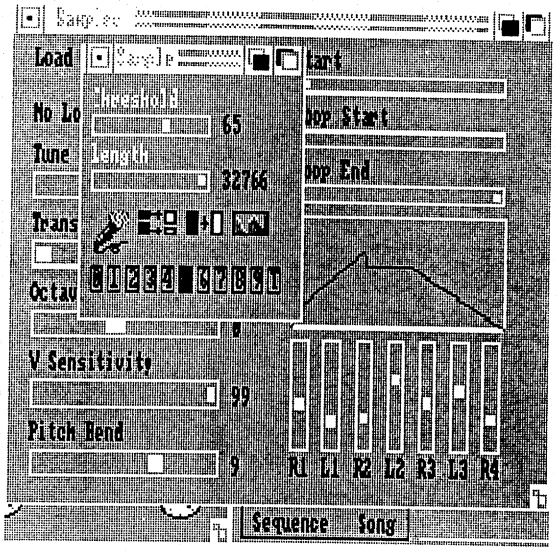


Figure 6: The Sample window.

THE PLAYER PIANO WINDOW

To open the Player Piano window (shown in Figure 7) and use the Player Piano module, you can connect any input module icon in the Patch Panel to the Player Piano icon, or you can double-click on the Player Piano icon. Try connecting the Console Keyboard icon to the Player Piano. The Player Piano window will open, and you can see it in action. Whenever you press any of the playing keys on the Amiga's keyboard, the Console Keyboard module sends notes to the Player Piano module, which plays the notes as flashing keys in its window. The Player Piano can take notes from other input modules as well and display them as flashing notes to provide a visual check on music information passing through the Patch Panel.

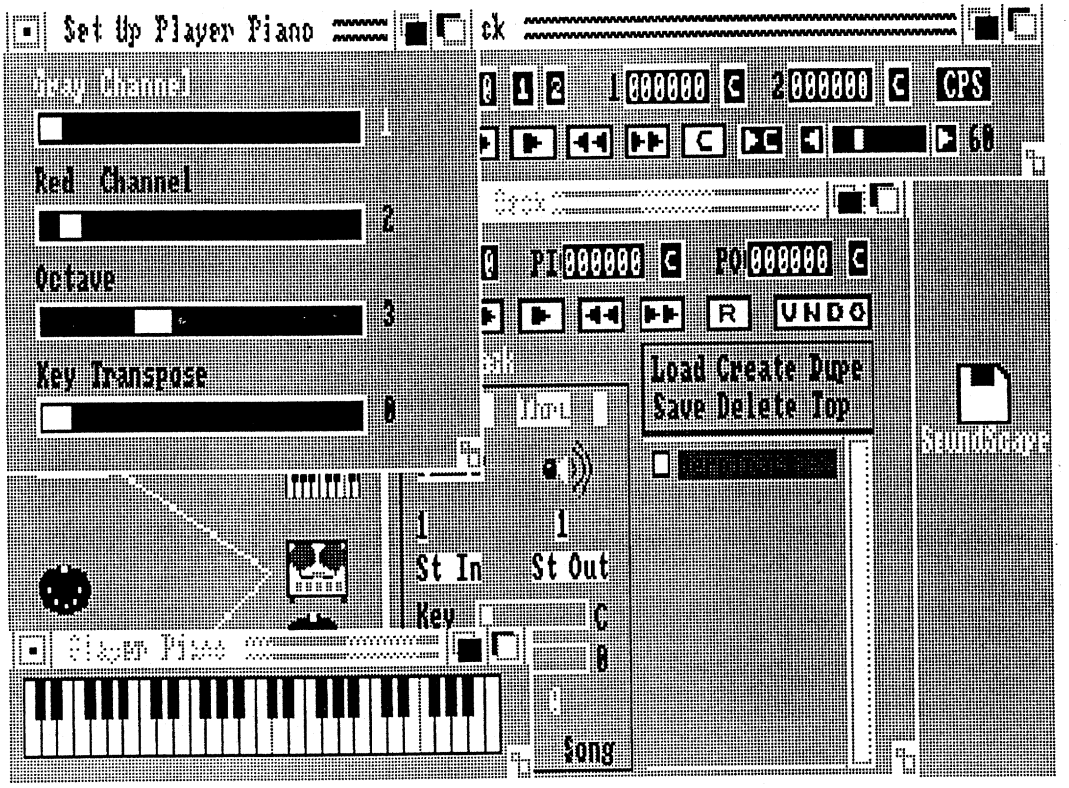


Figure 7: The Player Piano and Set Up Player Piano windows.

THE SET UP PLAYER PIANO WINDOW

If you double-click on the Player Piano icon, the Set Up Player Piano window (shown in Figure 7) opens along with the Player Piano window. You can control the way the Player Piano reacts to incoming music data by setting the sliders in this window. When you're done looking at this window and the Player Piano window, close them to clear up the Workbench.

THE CLOCK WINDOW

The Clock window (shown in Figure 1 at the beginning of Chapter One) appears anytime you connect the Clock module to another module, or when you double-click on the Clock icon in the Patch Panel. It also opens automatically whenever you open the Tape Deck window. You already used the controls in the Clock window to play a song, so you have some idea of what it does; it controls other modules by sending out control and timing signals to them. When you played a song, the Clock was sending control and timing signals to the Tape Deck. The Clock can also send these signals to other modules. The counters across the top of the window keep track of the timing signal count. The line of buttons across the bottom of the window turns the clock on and off and resets the counters. The slider in the lower right of the window controls the speed of the clock.

THE TAPE DECK WINDOW

SoundScape's Tape Deck module has more features than any other SoundScape module, so the Tape Deck window (shown in Figure 1 at the beginning of Chapter One) has a wide variety of controls and commands that open other windows. At the top of the Tape Deck window is a set of counters and buttons, much like the Clock controls, that allow you to stop and start the Tape Deck independently of the Clock controls. Below the buttons on the right side of the window is a rectangular area that lists the separate tracks in use on the Tape Deck. Each track can record musical information coming into the Tape Deck from outside modules. The commands above the track area create, duplicate, and otherwise manipulate tracks.

To the left of the track area is another rectangular area that contains controls for setting the recording and playback characteristics of any single track. Several of these controls let you connect an individual track directly to other SoundScape input and output modules without using the Patch Panel.

THE STATUS WINDOW

If you click on the St In command in the middle left of the Tape Deck window, a window titled "Status" (shown in Figure 8) will open up. This window contains a list of six different kinds of music data that can come into a Tape Deck track. By clicking on the data names, you can let different kinds of data pass into the track or block them from the track (see "Using MIDI Filters" in **Recording Scores.**) The St Out button next to the St In button in the Tape Deck window will also open the Status window if you click on it, but the controls in this case will set the different kinds of data that can pass *out* of the track instead of into the track. Close the Status window when you're done looking at it.

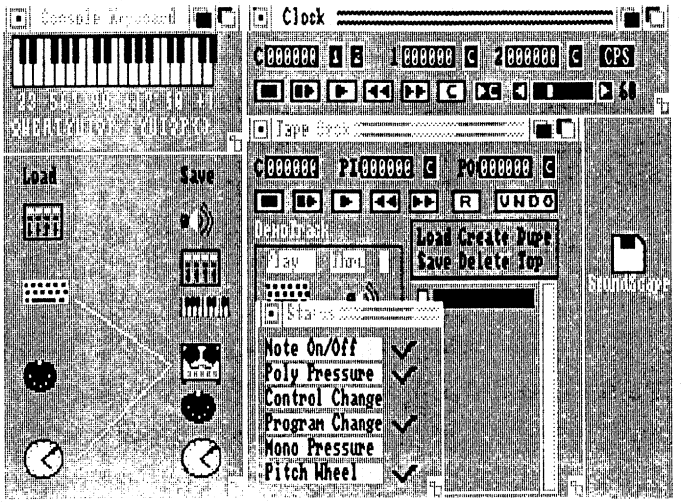


Figure 8: The Status window.

THE EDIT SEQUENCE WINDOW

If you click on the Sequence command in the lower left corner of the Tape Deck window, the Edit Sequence window (shown in Figure 9) will open. The large rectangle on the right side of the window lists a sequence of the music events like notes and special effects that are stored in a single Tape Deck track. The commands and controls to the left of the list let you add, delete, alter, and edit the events in the sequence so you can change the way they

sound when you play the Tape Deck. You can also use the commands to save and load sequences of events to disk. When you're done looking at the window, close it.

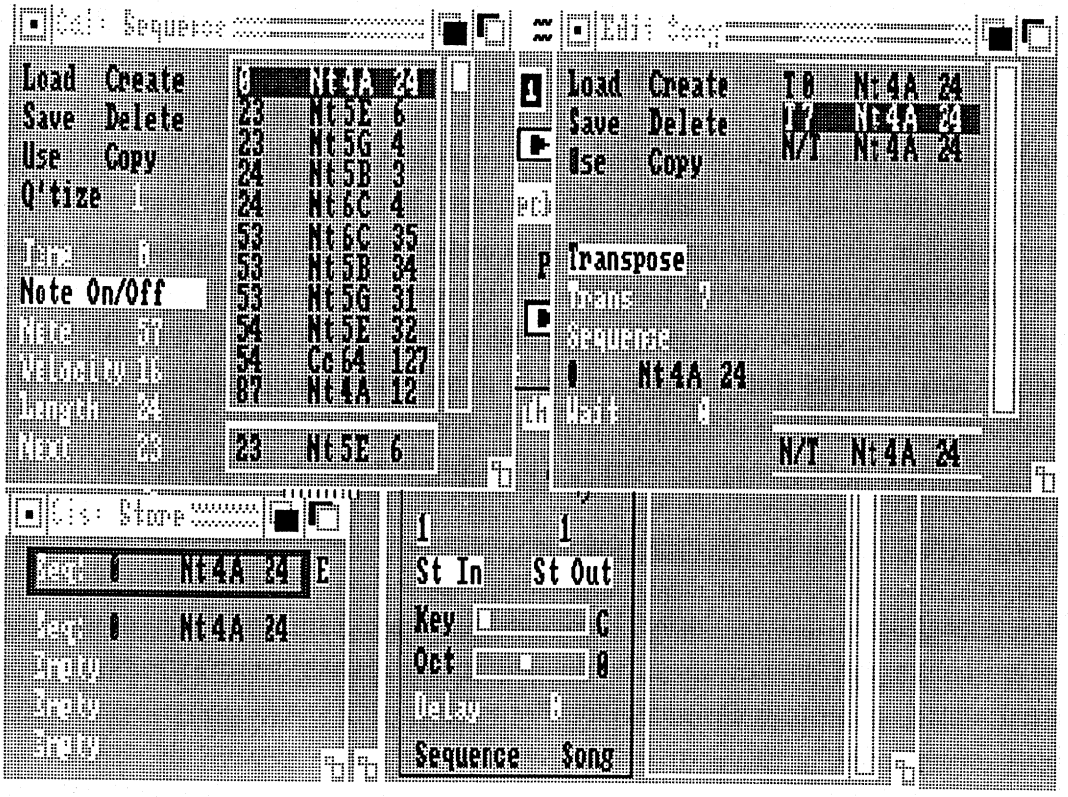


Figure 9: The Edit Sequence, Edit Song, and List Store windows.

THE EDIT SONG WINDOW

Next to the Sequence command in the Tape Deck window is the Song command. If you click on it, the Edit Song window (shown in Figure 9) will open. It looks much like the Edit Sequence window, but instead of listing individual music events in the right of the window, it lists full sequences of events. The Edit Song window lets you build songs with the commands and

controls in the left of the window by combining and repeating different sequences. For example, you might want a song with different verses, a bridge, and a repeating chorus. If you record the different verses as different sequences, the bridge as a separate sequence, and the chorus as yet another sequence, you can tie them all together in the Edit Song window to create a full song.

THE LIST STORE WINDOW

If you click on any one of the sequence listings in the Edit Song window, a small window titled "List Store" (shown in Figure 9) will appear. The List Store has five different slots that you can use to store sequences from the Edit Sequence window or songs from the Edit Song window. If you're familiar with word processors, this window is like a word processor's "cut and paste" buffer: you can cut a sequence or song from one source, store it here, and then paste it in another location. When you're done looking at the List Store window, close both it and the Edit Song window to clear up the Workbench.

SAVE AND LOAD REQUESTERS

Throughout the different SoundScape windows are Save and Load commands that will save and load environments, instrument banks, scores, tracks, songs, instruments, and other SoundScape data from disk. Anytime you click on these commands, a Save or a Load requester (both shown in Figure 10) opens. These requesters look and work the same wherever they appear, but save and load different kinds of data depending on which window you're in when you choose Save or Load.

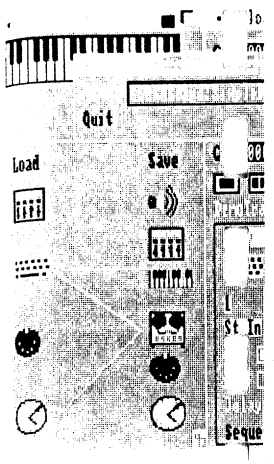


Figure 10: T

ACTIVE

Each time you
active--you a
most windows
have to do i g
Setting the cor
you're settir
button in the C
in the Clock wi

There are a few
are open. The
before you ca
window to wor
window, th
Sequence winc
using the T p
window. If yo
then go back t
to free up th c

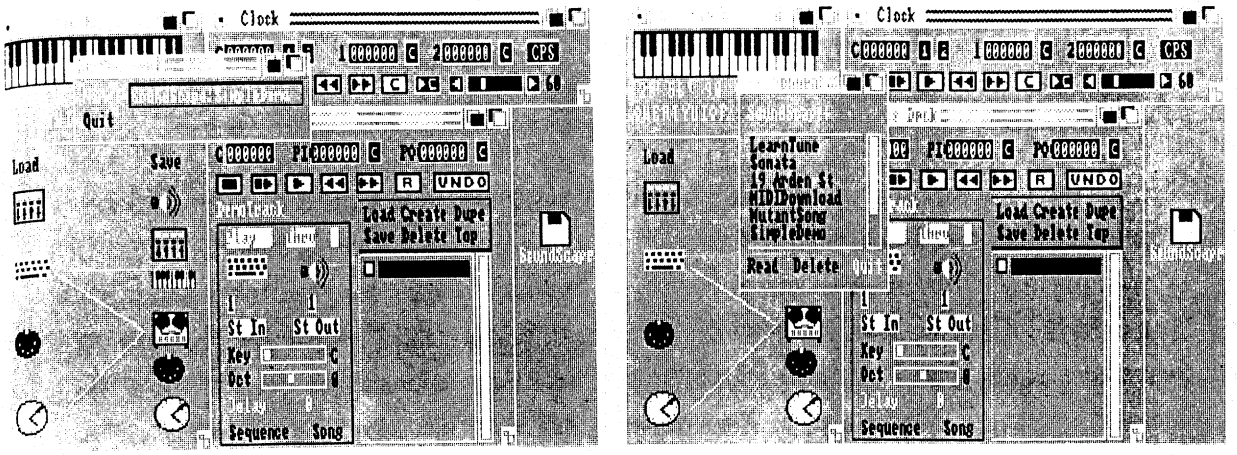


Figure 10: The Save and Load requesters.

ACTIVE WINDOWS

Each time you open a SoundScape window, that window is selected and active--you can move its controls and control the window's module. With most windows, if you want to set the controls in another window, all you have to do is go directly to the controls with the mouse pointer and set them. Setting the control automatically makes the window active. For example, if you're setting controls in the Tape Deck window and then want to push a button in the Clock window, all you have to do is click directly on the button in the Clock window.

There are a few windows that won't let you use other windows as long as they are open. These windows usually represent an activity that must be finished before you leave them. For example, if you open up the Edit Sequence window to work with a sequence and try to set controls back in the Tape Deck window, the Tape Deck controls will have no effect as long as the Edit Sequence window is open, even if you can move the controls. To get back to using the Tape Deck window, you have to first close the Edit Sequence window. If you ever find you can't use the controls in a SoundScape window, then go back to the window you were working in last and close that window to free up the controls.

ADDING MODULES

Now you've seen all the different windows and modules found in SoundScape Pro. Keep in mind that SoundScape is an open ended music operating system; you can add additional modules to it as Mimetics and other software companies release them on new disks. These additional modules will show up as additional icons on the Patch Panel, and as additional windows with controls on the Workbench. The new modules will work and exchange data with all the other existing SoundScape modules.

§§§§§§§§§§§§§§§§

n SoundScape
ı ting system;
her software
s will show up
vindows with
exchange data

Chapter Three: MIDI and SoundScape

SoundScape uses MIDI to transfer music data back and forth between its separate modules and between the Amiga and external music equipment you may have connected to it. If you already know what MIDI is, then the way SoundScape works with its Patch Panel and modules probably makes perfect sense to you. If you've never used MIDI before, a short explanation of MIDI will help you understand the inner workings of SoundScape and make clear what you're doing as you connect modules in the Patch Panel.

THE MIDI STANDARD

The first question most people ask when they encounter MIDI for the first time is: "What is a MIDI?" It's not quite the right question to ask, because MIDI isn't a thing; it isn't a piece of hardware or software. Instead, it's a set of common specifications that create a standard for communicating music data between different pieces of musical equipment.

The full name for the MIDI standard is "Musical Instrument Digital Interface". MIDI's specifications were agreed upon by various synthesizer, computer, and software companies so that products of one company can be easily connected with products of another company. With the MIDI standard,

there are infinite equipment combinations and musical possibilities. For example, musicians can play on the keyboard of one synthesizer that sends the played notes to a second synthesizer that actually sounds the notes. A series of synthesizers, drum boxes, and other electronic musical instruments can be connected together to play together in synchronization. Computers can send out music data to play music on attached synthesizers, and can record human performances on attached keyboards. These are just a few of the possibilities.

Part of the MIDI standard specifies physical connections: how many wires there are in a MIDI cable, what kind of ports are used in an instrument to accomodate cables, how fast data is sent over a cable, and other specifications. The other part of MIDI is the format of transmitted data: what kind of events can be transmitted, what codes stand for what events, and so on. By learning a little of both the hardware and software sides of MIDI, you can get a good understanding of how MIDI does what it does.

PHYSICAL CONNECTIONS

Although the MIDI specs go into great detail specifying the hardware that's used to connect different musical equipment, actually connecting equipment is quite simple. Each piece of equipment you want to connect is called a *MIDI device* . The cables you use to connect the devices are *MIDI cables* . Each MIDI cable sends a single stream of music data in one direction only. If you want to connect two MIDI devices so they can talk back and forth to each other, you must use two MIDI cables: one so the first device can send music data to the second device, the other so the second device can send music data to the first device.

MIDI allows you to connect together a chain of devices if you wish. In a device chain, the first device is usually the master device, a device that sends out data to and controls all the other devices. A cable from the master device to the second device sends data to the second device. Another cable from the

second device to the third device in the chain passes on the data from the master device to the third device. The third device passes the same data on to the fourth device through another cable, and so on down the line until the data reaches the last device in the chain.

MIDI MESSAGES

The MIDI cables connecting MIDI devices carry music data in the form of *MIDI messages* . Each MIDI message is a block of data that communicates a single music event. Music events can be things like the beginning of a note, the end of a note, or a change in the control settings of a synthesizer. For example, consider a musician playing a synthesizer that sends MIDI data out over a cable. When he presses a key, the synthesizer sends a *Note On* MIDI message over the cable that says that a key was pressed and tells the pitch of the key. When he releases the key, the synthesizer sends a *Note Off* message that says that the key was released. If he turns the pitch wheel on the synthesizer, the synthesizer sends a stream of MIDI messages that tells how far the wheel is turned as long as he keeps the wheel turned.

When the MIDI messages reach a receiving MIDI device, the device can act on them according to its own characteristics and settings. For instance, if a second synthesizer received the MIDI messages from the last example, it could sound the notes played from the first synthesizer using its own sounds. If the second synthesizer couldn't bend pitch, then it would ignore the pitch wheel messages it receives.

MIDI Channels

When a master device in a MIDI chain sends messages out to the other devices in the chain, all the devices in the chain can receive those messages, since all devices receive the same stream of signals from the master device. If all the devices in the chain respond to all of the master device's messages, they would play in unison, and the master device wouldn't be able to control any

device individually. The MIDI standard solves this problem by assigning messages to different *MIDI channels* .

To understand how MIDI channels work, think of the master device as a television transmitter that transmits over many different TV channels. All the other MIDI devices in the chain are homes with TV sets. If each home tunes in to one TV channel, the home receives only the messages transmitted on the channel they're tuned in to. If each home is tuned in to a different TV channel, then the transmitter can send a message to one house without sending it to the other houses by sending the message over the one TV channel that the house is tuned into.

MIDI uses 16 different channels to send messages. MIDI devices can usually be set to send or receive messages on any one of the 16 channels. In a MIDI chain, you can set each of the devices to receive a different MIDI channel, and then use the master device to send messages to an individual device by sending it only on the channel that the device receives. If you want several devices to respond in unison, then you can set them to receive messages on the same channel.

System Messages

Messages that travel over a single MIDI channel are called *channel messages* . Only the devices tuned in to that channel receive the channel messages. There are other types of MIDI messages that are received by all connected devices no matter what channel they're tuned in to, a sort of "all channels bulletin". These messages are called *system messages* .

System messages are usually messages that help to synchronize different devices so they start and stop playing music together. Typical system messages are *Start* , *Stop* , and *Reset* . The most commonly used system message is the *timing clock* (sometimes called a MIDI clock). Timing clocks

are messages that are sent at a regular interval over the MIDI connections, and are used like a metronome to set the speed of music.

Timing clocks are usually used for devices like drum boxes or sequencers, devices that have preset or prerecorded rhythms and notes to play. When these kinds of devices receive clocks, they play back their rhythms at the speed they receive their clocks. The MIDI standard is 24 timing clocks per quarter note, so if, for example, a drum box receives 24 clocks per second, it would play its rhythm back at one quarter note per second (a metronome marking of quarter equals 60, or 60 beats per minute). Synthesizers don't usually need timing clocks, since they don't often use preset rhythms.

HOW SOUNDSCAPE USES MIDI

Most MIDI programs for computers turn the computer into a single MIDI master device that sends and receives data from other MIDI devices attached to it. SoundScape is different. Since it runs on the Amiga computer, it can use the Amiga's special multi-tasking abilities to run many different music programs at once, effectively turning the Amiga into several MIDI devices all contained in one computer. These multi-tasking programs are the modules you saw in the quick tour of SoundScape.

Like regular MIDI devices, SoundScape modules must be connected to each other to be able to work together. You can make connections in the Patch Panel, where a white line appears to show connections between modules. These connections between modules are the equivalent of MIDI cables between MIDI devices: They send music data in one direction only, from the modules in the left column to the modules in the right column. If a module can both send and receive MIDI messages, then it has an icon in each column so you can make connections that either feed messages into the module, or receive messages coming out of the module.

Like different MIDI devices, each of SoundScape's modules has its own characteristics, and deals with MIDI messages in its own way. What follows is a description of each SoundScape module and how it passes MIDI information back and forth with the other SoundScape modules.

THE CONSOLE KEYBOARD

The Console Keyboard sends MIDI messages for Note On and Note Off on any one of the 16 MIDI channels. It sends timing clocks and control messages over all channels (since they're system messages). It doesn't receive MIDI messages. To send Note On and Note Off messages, you press any of the playing keys (QWERTYUIOP[] and 2356790=\). To send timing clocks and control messages, you use the function keys and the space bar. The Console Keyboard is the equivalent of the keyboard on a MIDI synthesizer; it turns your performances into note data and control signals.

THE SAMPLED SOUND PLAYER

The Sampled Sound Player receives MIDI messages on any one of the MIDI channels, but doesn't send messages out. It turns the messages it receives into sound using the Amiga's internal sound chips. Since the Sampled Sound Player responds to many different kinds of channel messages, it will play notes using different pitches, will play notes with varying loudness, and will bend the pitch of notes up or down when it receives pitch wheel messages. The Sampled Sound Player is the equivalent of the oscillators in a synthesizer; it takes note data from a source like a keyboard and turns it into sound.

THE CLOCK

The Clock can both send and receive MIDI messages. Since it only deals with system messages like stop, start, and timing clocks, it doesn't use MIDI channels. The Tape Deck is the only module in SoundScape that uses the Clock's messages, since it is a sequencer and recorder that needs timing

references. If you connect the Clock to external drum boxes or sequencers through the MIDI Out module, the Clock can control those devices with its messages.

If you connect an external timing device to the Clock through the MIDI In module, then the Clock can receive MIDI system messages like Stop, Start, and Reset that do the same work as the Clock's control buttons and run the Clock as if the external timer was a remote control.

THE TAPE DECK

The Tape Deck is SoundScape's most powerful module. It sends and receives MIDI messages on all 16 channels, and can use all channels simultaneously if you want it to. To keep the different channels and channel arrangements separate in recording, the Tape Deck uses *tracks* . Each track has its own distinct MIDI channel and can be connected to its own input module or output module, much like the tracks on a multi-track tape recorder can record different audio signals coming from different musical instruments.

The Tape Deck's main purpose is to record incoming MIDI messages on its tracks, and to play them back later just the way they were recorded. Again, it does this much like a standard tape recorder, recording on multiple tracks simultaneously when you record, playing back multiple tracks when you play back your recording. The Tape Deck uses the timing clocks it receives from the Clock module as a timing reference while it records so it knows how much time elapses between each MIDI message it receives. It also uses the timing clocks to control its playback speed--you can speed up or slow down playback by changing the rate of the timing clocks that the Clock puts out.

One thing the Tape Deck does that a standard tape recorder doesn't do is let you look at the contents of any track. By opening an editing window like the Edit Sequence or Edit Song window, you can see every event that was recorded in a track, and then alter them or add and delete events. This

changes the results when you play back a recording. With careful work in the editing windows, you create entire sequences of MIDI messages without actually recording anything.

THE PLAYER PIANO

Like the Sampled Sound Player, the Player Piano can receive MIDI messages but doesn't send messages. Instead of turning the MIDI messages it receives into sound, it turns them into a video display. It can receive messages on any two different MIDI channels simultaneously. It displays notes it receives on the first channel as flashing blue piano keys, and notes it receives on the second channel as flashing red piano keys. There's no direct equivalent to the Player Piano in the traditional music world, so you might think of it as something like a VU meter on a tape recorder: it shows whether or not you're getting a signal on any two MIDI channels.

THE MIDI MIXER

The MIDI Mixer can receive MIDI messages on all 16 MIDI channels and also sends messages on all 16 channels, so it has two icons on the Patch Panel; one for input, the other for output. If you connect a SoundScape module that sends messages on a single MIDI channel to the MIDI Mixer input, the MIDI Mixer will send the same messages out through the same channel in its output. You can think of the MIDI Mixer as a conduit filled with 16 connecting cables for the 16 different channels, with ports on each end that let you plug any number of modules to any one channel.

You can use the MIDI Mixer to combine the output of two different modules by setting the modules to send on the same MIDI channel, plugging both modules into the MIDI Mixer input, and then plugging the output into the source you want the modules to go to. However, you can do the same thing by connecting both modules to the same module in the Patch Panel. The main purpose for the MIDI Mixer is to work with Tape Deck tracks. You don't

connect individual tracks to SoundScape modules in the Patch Panel; instead, you choose one input and one output module for each track in the Tape Deck window. If you choose the MIDI Mixer as the track's input or output module, then you can connect several different modules to the Mixer in the Patch Panel. The track then sends and receives messages from multiple sources instead of just one single source.

THE MIDI IN MODULE

The MIDI In module takes any MIDI messages the Amiga receives through its MIDI In port and passes them on to SoundScape's Patch Panel so they can be sent to any one of SoundScape's modules. In this way, MIDI In provides incoming signals from external MIDI devices. It can also shift the channels of incoming messages to new channels so that you can set incoming messages to the channels you want to use in SoundScape.

THE MIDI OUT MODULE

The MIDI Out module takes any MIDI messages that come into it through the Patch Panel and passes them out through the Amiga's MIDI Out port to attached external MIDI devices. The MIDI Out module, along with the MIDI In module, is SoundScape's connection to the real world outside the Amiga.

These are SoundScape's modules, and this is the way they work use MIDI messages to work with other modules and external devices. By learning to use them effectively, you can twist, torture, and tease MIDI data until you get just the results (and music!) that you want. The next section of this manual will show you how to do just that.

§§§§§§§§§§§§§§§§

Section Two:

A SoundScape Tutorial

Play

The two Sou d
Amiga's own n
Sound Player m
using examp s
You'll also lear
your own sa p
the SoundScape

USING TH

You'll recall fr
keystrokes o
messages to ot
Keyboard m lu
time recordin
section of the m
generating ab i

To use the C n
Keyboard ico
another windo
Workbench

Playing Sampled Sounds

The two SoundScape modules that take the most direct advantage of the Amiga's own hardware are the Console Keyboard module and the Sampled Sound Player module. In this chapter, you'll learn how to use both modules, using examples that show you how the two modules can work together. You'll also learn how to use the Sampled Sound Player to make and modify your own sampled sound instruments. In addition, you'll learn how to use the SoundScape Save and Load requesters.

USING THE CONSOLE KEYBOARD MODULE

You'll recall from Section One that the Console Keyboard module turns keystrokes on the Amiga's keyboard into note information, sent as MIDI messages to other SoundScape modules. You can also use the Console Keyboard module to send timing clocks, an advanced feature used for step time recording on the Tape Deck, a feature covered in a later chapter. In this section of the manual, you'll learn to work with the Console Keyboard's note generating abilities and leave step time recording for later.

To use the Console Keyboard module, first double-click on the Console Keyboard icon in the Patch Panel. The Console Keyboard window and another window titled "Edit Console Keyboard" will open up on the Workbench if they aren't open already. The Console Keyboard window

shows you how pitches are arranged on the Amiga's keyboard. If you select this window, you can see the pitches you play on the Amiga keyboard flash on the window's keyboard. If you want to change the range of the pitches the Amiga's keyboard plays, you can move them up by an octave: Press the CAPS LOCK key so its light glows. The keyboard will play pitches an octave higher than it did before. Press the CAPS LOCK key again so the light goes out, and the pitches will drop back down to the original octave.

SETTING THE EDIT CONSOLE KEYBOARD WINDOW CONTROLS

As you play notes on the console keyboard, the Console Keyboard module reads them and sends them to the Patch Panel as MIDI messages. If you want to change the way the Console Keyboard module sends out MIDI data, select the Edit Console Keyboard window. There are three sliders here you can use:

- The **Channel slider** sets the MIDI channel that the Console Keyboard uses to send out MIDI data. You can set it from channel 1 to channel 16.

- The **Octave slider** sets the octave range of the pitches that the Console Keyboard sends out. It's normally set to octave four, but you can set it from octave 0 to octave 8. Octave 5 is the octave that contains middle C. By setting this slider, you can play a very wide range of pitches with the Console Keyboard.

- The **Key Transpose** slider transposes the pitches that the Console Keyboard sends out by 1 to 11 half-steps up. By setting this slider, you can make the QWERTYUIOP[] row of console keys play a scale in any key you want.

Under the three sliders are six boxes outlined in red. Each box is preceded by a label in black, and contains a value in white. The labels F1 through F5

stand for the first five function keys on the Amiga's keyboard. The label SB stands for the space bar. These boxes control how many timing clocks function keys 1 to 5 and the space bar on the Amiga's keyboard send out when you press them. To change the contents of any box, click on the box so a cursor appears, then use the Amiga's editing keys (the cursor keys and the DEL and BACKSPACE keys) to change the value to another value. You can read about using the function keys and the space bar in step time recording in the "Recording Scores" chapter later in the manual.

When you're done setting sliders and key values, close the Edit Console window by selecting the window's close gadget.

USING THE CONSOLE KEYBOARD MODULE WITH THE SAMPLED SOUND PLAYER

The Sampled Sound Player uses sampled sound instruments to play notes it receives from an incoming MIDI signal using the Amiga's sound synthesis hardware. It can hold up to 16 different instruments at a time, one for each of the 16 different MIDI channels it can receive MIDI data on. It can also create sampled sound instruments using the SoundScape Sampler, edit sampled sound instruments, and save and load sampled sound instruments from disk.

You can play the Sampled Sound Player by connecting any of the input module icons on the right of the Patch Panel to the Sampled Sound Player icon. The Sampled Sound Player will play the notes it receives from the connected input module (or modules). The Console Keyboard module is a very convenient input module to use as an input module--you can send MIDI notes to the Player by playing the Amiga's own keyboard--so this example and most of the other examples in this manual use the Console Keyboard module to generate notes. If you have an external MIDI keyboard that you'd

rather use, you can connect it to the Amiga with a MIDI adaptor and cable, and then use the MIDI In module to send notes from the external keyboard to the Sampled Sound Player or other SoundScape module. (You'll find more information about the MIDI In module in the last chapter of this manual.)

To use the Sampled Sound Player with the Console Keyboard, follow these steps. They show you how to load instruments in the Sampled Sound Player so you can play them:

1. Connect the Console Keyboard icon (or the MIDI In icon if you plan to use an external MIDI keyboard) to the Sampled Sound Player icon in the Patch Panel by first clicking on the Console Keyboard icon and then clicking on the Sampled Sound Player icon. A line will connect the two icons.

2. Double-click on the Sampled Sound Player icon to open the Sounds window. The Sounds window will open displaying 16 numbered lines of text. These lines show the names of the sampled sound instruments loaded in each of the Player's 16 instrument channels. If the Player receives notes on MIDI channel 1, it plays them using the instrument loaded in instrument channel 1, if it receives notes on channel 2, it uses instrument 2, and so on for each of the 16 channels. If you haven't loaded an environment already, the lines should now all read "No Sound", which means there are no sampled sound instruments loaded in any of the Player's 16 instrument channels.

3. Double-click on line 1 to load an instrument in channel 1. The Sampler window opens. The controls in this window let you change or edit a sampled sound instrument.

4. To load a sampled sound from disk, select the Load command in the upper left corner of the window. A load requester opens.

5. The load requester shows a list of all the sampled sound instruments on a disk in the list box in the center of the requester. The list box shows the

instruments found on whatever directory is specified in the directory name at the top of the requester. If there are no instruments in the list box, click on the directory name and press RETURN to load the instrument names from disk. (If there are no instruments on the disk, you still won't see any instrument names.) If you want to see instruments on a different disk drive, click on the directory name to make a cursor appear, then change the directoy name using the Amiga's editing keys. The directory names below choose different disk drives:

DF0: The internal disk drive
DF1: The first external disk drive
DF2: The second external disk drive
DF3: The third external disk drive
DH0: The hard disk drive

Press RETURN when you have the directory name you want, and the list box will show a new list of the instruments on the disk in the disk drive you specified. If you want to look on a different disk using the same disk drive, first eject the disk in the drive and insert the new disk, then select the directory name and press RETURN without changing the directory name. A new list will appear in the requester.

You can use the directory name to enter pathnames to get to any directory on any disk drive or memory device currently attached to the Amiga. For more information about pathnames, consult the *AmigaDOS User's Manual* .

6. If there are more instruments on the disk than the list box can show, you can scroll through them using the scroll bar on the right of the box. It works like a slider; drag it up and down to see different parts of the instrument list. To choose an instrument, click on the instrument name, then select the Read command. The window closes and the instrument loads. As a shortcut, you can double-click on the instrument name and the

instrument will load without selecting the Read command. Select Quit if you want the instrument requester to close without loading an instrument.

7. You can hear the instrument by playing the top two rows of keys on the Amiga's keyboard: QWERTYUIOP[] and 23 567 90 =\. If you don't hear any sound (and your monitor speaker is turned up), select the Console Keyboard window. The console keyboard can always play if the Console Keyboard window is selected. If you select other windows that use the keyboard for entering characters, the Console Keyboard won't play notes.

One note about Load Requesters: the first time you load the contents of a directory to display in the requester, SoundScape compiles the list directly from a disk. To save you time later, SoundScape saves the contents of the list in memory so that when you open the requester a second time the contents appear immediately without a disk referral. If you save new instruments or other files to disk, they won't show up in the list unless you click on the directory name and press the RETURN key to load the list again directly from disk.

Once you've loaded an instrument, you can close the Sampler window and the Sounds window and the Sampled Sound Player will still play when you use the Amiga keyboard. If you want to load more instruments in the different instrument channels, you can open the Sounds window and repeat the loading process for each channel you want to load. But be warned: Samples take up a lot of memory and it's very easy to use it all up with a few channels of big fat sounds. This leaves no room for all kinds of other things, like the Tape Deck, or the *Amiga* Operating System, to do their business. When memory is running low, you will be warned with a message.

CREATI(INSTRUMI

If you have a mi
your Amiga thr
sampled soun ii

1. Choose one of
the last exa p

2. Select the S n
opens up.

3. Set the Ler t
The Length s
recording. T
maximum s t

4. Set the Th s
SoundScape
(SoundScaj :
waits for a r

5. The 10 nur)(
can record in.
to record in (

6. Click on the n
waits for a g
before it starts
when it's fir sl

7. To test the sa
Keyboard i c

CREATING YOUR OWN SAMPLED SOUND INSTRUMENT

If you have a microphone, tape deck, or other source of sound connected to your Amiga through the SoundScape Sampler, you can create your own sampled sound instruments by following these steps:

1. Choose one of the Sampled Sound Player instrument channels as you did in the last example so the Sampler window opens up.

2. Select the Sample command in the top of the window. The Sample window opens up.

3. Set the Length slider to set the length of the sampled sound you record. The Length slider sets the amount of memory in bytes available for recording. The sampler uses about 15,000 bytes per second, so the maximum setting of 32,766 bytes gives you about two seconds of time.

4. Set the Threshold slider to set the incoming audio signal strength SoundScape waits for before it begins to sample. It ranges from 0 (SoundScape starts sampling as soon as you ask it to) to 100 (SoundScape waits for a *very* loud sound level before it starts sampling).

5. The 10 numbers across the bottom of the window are the 10 octaves you can record in. Click on one of the numbers to choose the octave you want to record in. (4 and 5 are the octaves the Console Keyboard usually plays.)

6. Click on the microphone icon to start the sampling process. SoundScape waits for a signal stronger than the value you set in the Threshold slider before it starts sampling. The screen blanks as it samples, and comes back when it's finished.

7. To test the sample, play the Amiga keyboard. (Make sure the Console Keyboard is connected to the Player and is sending notes on the channel

that you're using and in the octave that you recorded in. Click in the Console Keyboard window if you're not hearing any notes.) To hear the sample in the same pitch it was recorded, play C (that's 'I' on the console keyboard.)

TRANSLATING ONE OCTAVE SAMPLE INTO OTHER OCTAVES

The sample you make will play within a one octave range from A natural to G sharp above it. If you want to use other octaves to record other samples, click on a new octave number and go through the sampling procedure again. If you want to translate one sample up or down into other octaves so that one sample will play over a range of octaves, follow these steps:

1. Click on the octave with the sample you want to translate. The box turns red.

2. Click on the octave you want to translate to. The box turns white.

3. Click on the icon just right of the microphone icon (it shows two small white boxes going to two small red boxes). The sample translates from the original octave (red) to the new octave (white).

COPYING ONE OCTAVE SAMPLE INTO ANOTHER OCTAVE

If you want to copy the contents of one octave directly to another octave without translating it up or down (so the contents of the copied octave will play in the same pitch range as the original octave), follow the same procedures you just used to translate an octave, but in step 3, click on the second icon to the right of the microphone instead of the first icon to the right. (It shows one white box with an arrow to one red box.) SoundScape will make an exact copy of the contents of the original (red) octave in the duplicate (white) octave.

MOVING SAMPLES BETWEEN CHANNELS

In addition to copying samples from one octave to another, you might find it useful to copy samples from one channel to another. This can be done by copying the sample into the 'T' register. Then leave the Sample window and the larger Sampler window, select another channel and open its Sample window. Octaves 0 through 9 will have changed, but 'T' will be just as you left it. Copy 'T' into whichever octave you'd like.

COMPRESSING SAMPLES

The rightmost icon in the row of icons in the Sample window is the Compress/Non-Compress switch. By clicking on it, you can switch back and forth between Compress, which looks like a flat line across the icon, and Non-Compress, which looks like an audio wave with full peaks and valleys.

The Compress/Non-Compress switch works something like automatic level control on a tape recorder. If you set it to Compress, SoundScape compresses the incoming audio signal as it records it; it removes volume changes in the signal by boosting very soft sounds and cutting very loud sounds so the recording has a smooth overall level. If you set it to Non-Compress, SoundScape doesn't compress the audio signal, so the recording has a wide range of dynamics, from very soft to very loud if the original sound ranges from very soft to very loud. You can use Compress mode when you want to keep your samples from overloading with very loud sounds, or from not picking anything up when you try to record very soft sounds.

If you want more information on using the SoundScape Sampler to create new sampled sound instruments, you'll find it in the SoundScape Sampler manual. When you're finished with the Sampler, close the Sample window to go on to other examples.

EDITING A SAMPLED SOUND INSTRUMENT

If you want to change the sound or playback characteristics of a sampled sound instrument, use the controls in the Sampler window. Some of these controls alter the actual sound of the instrument, the others change the way the instrument reacts to incoming MIDI data. As you change the controls, you can test their effect by playing a note on the Amiga's keyboard each time after you reset the controls. (You must have the Console Keyboard connected to the Sampled Sound Player for this to work.)

CHANGING THE INSTRUMENT'S SOUND

Two commands at the top left of the window change the instrument's sound by controlling looping and octave playback:

- The **Loop/No Loop** command sets the instrument to loop through its sample as long as SoundScape holds out a single note using that instrument, or it sets the instrument to play through its sample only once, ending the sound at the end of the sample no matter how long the note is held. Select the command to alternate between the two options.

- The **Low/High** command changes the octave that the sample plays back in. The Low setting plays through the sample an octave lower than the sound was sampled. The High setting plays through the sample at the original octave the sound was sampled in. Select the command to alternate between the two options.

The three sliders in the upper right of the window set the way the Sampled Sound Player plays through instrument's sample:

- The **Start** slider sets the starting playback point in the sample.

- The **Loop Start** slider sets the start of the loop in the sample.

• The **Loop End** of the slider sets the end of the loop in the sample, and also sets the end of the sample if the Loop/No Loop command is set to No Loop.

The length of each slider represents the length of the sample, stretching from the beginning of the sample on the left end of the slider to the end of the sample on the right end of the slider. Drag the slider to the position in the sample where you want to set the loop or playback point.

The envelope display under these sliders shows the amplitude envelope that the Sampled Sound Player imposes on the sample playback, and changes to show the settings of the seven envelope sliders below it. You can think of the envelope as a contour used to shape each note the sampled sound instrument plays. Set the envelope sliders to change the envelope and the sound of the instrument. The sliders are divided into four rate sliders (R1, R2, R3, and R4), and three level sliders (L1, L2, and L3). The level sliders are:

• L1, the attack level: The level the amplitude goes to when a note first starts.

• L2, the decay level: The level the amplitude goes to after the note goes to L1.

• L3, the sustain level: The level the amplitude stays at as long as the note is held out .

The rate sliders are:

• R1: The rate the envelope uses to go from no volume at the beginning of the note to level 1.

• R2: The rate the envelope uses to go from level 1 to level 2.

• R3: The rate the envelope uses to go from level 2 to level 3.

• R4: The rate the envelope uses to go from level 3 to no volume at the end of the note (the decay rate).

As you set these sliders, the envelope display changes to show the new envelope. The length of the envelope stays the same within its box, so the time scale of the envelope change to reflect the proportions of the rate settings. For example, if you increase R1 to be much longer than R2, R3, and R4, then the first slope in the envelope will grow horizontally to take up most of the box, and the other three slopes will shrink (sometimes almost to nothing if the proportion is unequal enough.)

CHANGING THE INSTRUMENT'S PLAYBACK CHARACTERISTICS

The sliders on the left side of the window change the instrument's playback characteristics:

• The **Tune** slider fine-tunes the pitch of the instrument up or down by about a half-step in each direction.

• The **Transpose** slider transposes the playback of the instrument up by 1 to 11 half-steps from the pitch of the MIDI note it receives.

• The **Octave** slider transposes the playback of the instrument up or down by octaves over a range of 9 octaves from the pitch of the MIDI note it receives.

• The **V Sensitivity** slider sets the amount the instrument responds to velocity MIDI information it receives (from an external MIDI device) from 0 (no response) to 99 (full response). Velocity information usually comes from a touch sensitive keyboard that can tell how hard you press the keys. When the V Sensitivity slider is set to respond, the Sampled Sound Player will play the instrument louder as the keys are pressed harder.

• The **Pitch Bend** slider sets the amount the instrument responds to pitch wheel MIDI information it receives (from an external MIDI device) from 0 (no response) to 12 (a full octave of response). The pitch bend setting is in half-steps, and sets the range the instrument will bend up or down in response to the turn of a pitch wheel.

SAVING A SAMPLED SOUND INSTRUMENT

When you create or alter a sampled sound instrument, you can save it to disk by using the Save command:

1. Select the Save command at the top of the Sampler window. A save requester opens.

2. Click in the "Name:" box to get a cursor.

3. Use the Amiga's editing keys to change the name in the box to the name you want to save the instrument under. You can preface the name with a directory name (described earlier in the chapter) to save the file to a specific disk drive or directory on a disk.

4. Press the RETURN key to save the instrument. The requester disappears and the File Format? requester appears. (If you want to leave the requester without saving the instrument, click on the Quit command.)

5. Click on either Mimetics or IFF to save the instrument using either of these two standards. The requester disappears and SoundScape saves the instrument to disk.

The Mimetics standard saves the instrument with all its settings in the Sampler window and all the data in each of the sampled octaves. The IFF standard doesn't save all the Sampler window settings, and it trims some of the octave data. The IFF standard saves the instrument in a format that can be used by other Amiga software using the IFF standard (Deluxe Music and

Instant Music from Electronic Arts, for example). For more information on these two standards, read the SoundScape Sampler Manual included with the SoundScape Sampler.

CLEARING A SAMPLED SOUND INSTRUMENT FROM MEMORY

If you wish to clear the a sampled sound instrument from memory, choose the "Clear" command, located next to the "Sample" command. SoundScape will wipe the instrument's sample from the Amiga's memory, and leave you with a clear space for creating another instrument.

WORKING WITH BANKS OF INSTRUMENTS

As you load and create instruments in the Sampled Sound Player, the Sounds window will begin to fill up with instrument names in its 16 different slots. This collection of instruments is called an *instrument bank* . You can save and load the contents of the instrument bank by using the "Load All" and "Save All" commands at the bottom of the window.

To save an instrument bank, click on the "Save All" command. A save requester will appear that you can use to give your instrument bank a name and save it to disk. Whenever you save an instrument bank, you save all the instrument names in the Sounds window.

To load an instrument bank, just click on the "Load All" command. A load requester will appear listing the available instrument banks on the current disk. You can choose any one of the banks. SoundScape will load all the instruments listed in the saved instrument bank into memory so you can play them in the Sampled Sound Player. If you have the instruments stored in a separate directory from the instrument bank (instruments in a separate disk from the instrument bank for example), then the instrument bank won't be

able to load the instruments. You will have to use the Workbench to put the instrument bank and the instruments all in the same drawer so the instrument bank can load all the instruments.

§§§§§§§§§§§§§§§§

Playing Scores

When you played music in your introduction to SoundScape in Section One, you first loaded an environment, then you used the Tape Deck to play back the music. The environment you loaded took care of all the Patch Panel connections, the Sampled Sound Player instrument bank, and other SoundScape arrangements you needed for the music.

One of the main elements loaded from the disk with the environment was the *score* . A score is a collection of notes, pitch wheel information, and other MIDI data stored in the Tape Deck for playback. A score can be played back using the Sampled Sound Player, external MIDI synthesizers, or any other sound generating MIDI device capable of receiving and playing MIDI information. Although the environment you used to arrange SoundScape played the score using the Sampled Sound Player, you can set up the Tape Deck to play back a score in many other ways, depending on where you send the MIDI signals from the Tape Deck.

Setting up a score already saved on disk is an easy way to learn how to play back a score without using an environment. It's a simple matter of loading the score, setting the Tape Deck so it will send its signals out to a module like the Sampled Sound Player, and using the Tape Deck playback controls. In the process, you'll learn how to use the Clock, how to use playback controls, and how to work with Tape Deck tracks.

OPENING THE TAPE DECK WINDOW

To open the Tape Deck window if it isn't already open, double-click on the Tape Deck icon in the Patch Panel. The Tape Deck window and a second window, the Clock window, will open on the Workbench.

SoundScape automatically connects the Clock to the Tape Deck in the Patch Panel when you open the Tape Deck window because the Tape Deck won't work without the Clock. When a real tape deck records music, it records the notes on the tape that moves past its record head. When the real tape deck plays back the music, it can tell the order and duration of the notes it plays by their location on the tape. Since the SoundScape Tape Deck records notes in memory instead of on moving tape, it uses a stream of timing signals coming from the Clock module to keep track of time as it records notes so it can keep the notes in the proper order with the right durations.

USING THE CLOCK

When it's turned on, the Clock module sends out a steady stream of timing clocks to any modules it's connected to in the Patch Panel. The counter in the upper left of the Clock window counts the clocks as they go out. You can turn the Clock on and off, speed it up or slow it down, or reset the counter reading using the controls in the Clock window.

THE CLOCK BUTTONS

The seven buttons across the bottom of the window are from left to right:

- The **Stop button**, which stops the Clock and sends out a MIDI Stop message to attached modules.

• The **Go From Beginning button**, which sets the counter to 000000 and starts the Clock running. When you click on this button, the Clock sends out a MIDI Reset to 0 message to any attached modules before it starts sending the stream of MIDI clocks.

• The **Go button** starts the Clock running without resetting the counter.

• The **Rewind button** runs the counter backwards quickly as long as you point to the button and hold down the select button of the mouse. When you let up on the button, the counter stops where it is and the Clock sends out its current reading as a MIDI Song Pointer message to other attached modules. Rewind only works when the Clock is stopped with the Stop button.

• The **Fast Forward button** runs the counter forward quickly as long as you point to the button and hold down the select button of the mouse. When you let up on the button, the counter stops where it is and the Clock sends out its current reading as a MIDI Song Pointer message to other attached modules. Fast Forward only works when the Clock is stopped with the Stop button.

• The **Clock Lock button** locks the Clock on when the button is turned on so the Stop button has no effect on the clock's MIDI clock stream. The Stop button still sends out a MIDI Stop message to other modules, though.

• The **Clock On on Go button** enables the Clock to turn on when the Go or Go From Beginning button is pressed.

If neither the Clock On on Go button or the Clock Lock button is turned on, the Clock won't run at all.

THE TEMPO SLIDER

The slider to the right of the buttons is the **tempo slider** and sets the rate of the clock. You can drag the slider to the left to slow down the Clock or to the right to speed up the Clock. You can also click on either side of the little white box in the slider to make the slider jump in small increments to one side or the other. The two arrows on either end of the slider are for fine tuning. If you click on either arrow, the slider will move move a tiny increment towards the arrow.

THE CPS/BPM BUTTON

The number to the right of the tempo slider tells you the rate the Clock is sending out timing clocks. It can either show you the rate in clocks per second (CPS), or it can show you the rate in beats per minute (BPM). The button just above the number is the CPS/BPM button, and setting it determines the rate measurement. If you click on the button, it will change back and forth between CPS and BPM.

CPS is useful for dealing directly with MIDI clocks, but isn't always useful if you're used to using metronome markings. BPM is a standard method of measuring tempo found on most metronomes. In SoundScape, each beat measured using BPM is equal to 24 MIDI clocks.

THE AUTO-LOCATE REGISTERS

The two counters numbered 1 and 2 in the upper right of the window are the auto-locate registers. Whenever you click on the "C" button next to one of these registers, the register captures the exact reading of the Clock's main counter and stores it. If you stop the clock and click on the "1" button next to the main counter, the counter will jump to the location stored in auto-locate register 1. If you stop the clock and click on the "2" button, the main counter will jump to the location stored in auto-locate register 2. Any time you reset the Clock counter to an auto-locate location, the Clock sends the new auto-

locate setting out via MIDI messages to attached modules at the same time as it resets the counter.

CONTROLLING THE CLOCK FROM AN OUTSIDE SOURCE

The clock can receive MIDI messages as well as send them. If you connect an output module to the Clock's input icon in the Patch Panel, then the output module can start and stop the clock. For example, if you connect the Console Keyboard module to the Clock, you can start, stop, and reset the Clock using the Amiga keyboard's cursor keys (discussed in the chapter on recording with the Tape Deck) because the cursor keys send out MIDI Stop, Start, and Reset to 0 messages. You can also connect an external clock to the Clock through the MIDI In module so the external clock can control SoundScape's Clock with MIDI messages. MIDI drum machines are a good clock source.

USING THE TAPE DECK FOR PLAYBACK

When you look at the Tape Deck window, you can see a variety of different controls. Many of them are for recording and editing scores--you won't learn about them in detail in this chapter, but you can read about them in later chapters. The controls are grouped in four different areas:

- The **Tape Deck counters** across the top of the window.
- The **Tape Deck buttons** just under the counters.
- The **Track Control Box** in the lower left quarter of the window.
- The **Track List** in the lower right quarter of the window.

LOADING A SCORE IN THE TRACK LIST

The Track List shows the names of the tracks the Tape Deck is storing in memory. Each track stores one stream of MIDI signals, and can be connected to a different SoundScape output module. A score can have from one to as many tracks as memory will permit. If you haven't already opened an environment, when you first open the Tape Deck window, the Track List shows no tracks since there is no score loaded.

To load a score from disk, choose the Load command from the commands above the Track List. A load requester just like the one you used in the Sampler window will open displaying scores instead of instruments. Choose a score the same way you chose an instrument in the last chapter. When the score is loaded, you'll see the names of the tracks in the score displayed in the Track List. The top track is selected (highlighted in red). If there are more tracks loaded than the list can display, you can scroll through them with the scroll bar to the right of the list. You can select any of the tracks in the list by clicking on its name. The Track Control Box to the left of the Track List changes to show information for whatever track you select in the track list.

SETTING TRACKS FOR PLAYBACK

Before you play back a score, you should go through each track in the score to make sure it's set up for playback; when the score was saved, the tracks might have been set up to record or to play using an output module that you don't want to use, and they will still be set that way when you load the score back into memory. As you set the tracks for playback in this example, use the Sampled Sound Player as the output module. It's not the only output module you can use, but it is very convenient since you don't have to hook up any external equipment. To set up a score to play back using the Player, follow these steps:

1. Select the first track in the Track List if it isn't already selected.

2. The white rectangle in the upper left corner of the Track Control Box (called the operating mode button) should read "Play". If it doesn't, click on the button until it reads "Play".

3. The white horizontal rectangle in the upper right corner of the Track Control Box (called the playback mode button) should read either "Thru" or should be blank. If it's neither, click on the button until you get either "Thru" or a blank space.

4. The icon just beneath the playback mode button shows the SoundScape module that the track sends its MIDI signals to when the score plays back (the output module). The icon for this example should be the Sampled Sound Player. If it isn't, click on the icon to cycle it through the other available output modules until the Player icon appears.

5. The number just below the output module icon shows the number of the MIDI channel the track uses to send its MIDI signals on. If this is the first track, click on the number to cycle it through the channels until it's set on channel 1 (if it isn't already set to 1). If this is the second track, set the channel to 2, if it's the third track, set it to 3, and so on for all the tracks. You can set these channels to any number you want, but for this example it's convenient to match the MIDI channel number to the number of the track.

Repeat these steps for each of the tracks in the score. When you're finished, load a sampled sound instrument in the Sampled Sound Player for each one of the MIDI channels you just set up in the tracks for playback. These sampled sound instruments provide a voice for each of the tracks in the score. If you don't load an instrument for one of the channels used by a track, that track won't be able to play back.

USING THE TAPE DECK BUTTONS AND COUNTERS

Once you've set all the tracks in the score for playback, you can use the Tape Deck buttons and counters to play back the score. They work very much like the controls in the Clock window and the controls on a standard tape deck. The main counter in the upper left corner of the window measures how far the score has played back in MIDI clocks. The buttons are, from left to right:

- The **Stop button** stops the Tape Deck playback (or recording, if you're recording, as you will be in the next chapter).

- The **Play From Beginning button** sets the main counter to 000000 and starts the Tape Deck playing (or recording).

- The **Play button** starts the Tape Deck playing (or recording) without resetting the main counter.

- The **Rewind button** runs the main counter backwards quickly as long as you point to the button and hold down the select button of the mouse. When you let up on the button, the counter stops where it is. Rewind only works when the Tape Deck is stopped with the Stop button.

- The **Fast Forward button** runs the counter forward quickly as long as you point to the button and hold down the select button of the mouse. When you let up on the button, the counter stops where it is. Fast Forward only works when the Tape Deck is stopped with the Stop button.

- The **Record button**, when turned on, tells the Tape Deck to record on any tracks that are set to Record (covered in the next chapter). Has no effect on tracks set to Play.

• The **Undo button**, when pressed immediately after you record something, undoes the recording (covered in the next chapter).

The Play and Play from Beginning buttons will start the Tape Deck, but the main counter won't progress at all if you don't have the Clock running and connected to the Tape Deck. If the Clock is running and connected, you can use these two buttons to play back your score. Stop the playback with the Stop button. If you want to go to any particular part of the score, use the Fast Forward or Rewind buttons to advance or decrease the main counter reading, then use the Play button to start playback.

As you saw in the example in Section One, you can also control the Tape Deck using the Clock buttons, since the clock sends out Start, Stop, and Reset MIDI messages to the Tape Deck. The Tape Deck's own controls are useful if you have the Clock hooked up to other MIDI devices and you want to stop and start the Tape Deck without stopping and starting the Clock.

CHANGING THE PLAYBACK

You can change the way SoundScape plays back the score in several different ways:

1. You can speed up or slow down the score playback by setting the tempo slider in the Clock window.

2. You can change the instrument that each track plays by going to the Track Control Box and changing the MIDI channel that the track plays over, or by choosing a new output module. If you choose the MIDI Out module, you can play back the track using an external MIDI synthesizer.

3. You can change the instrument that each track plays in the Sampled Sound Player by loading different instruments for the MIDI channels.

4. You can tra p or the Oct slide half of the b c. by 0 to 11 half notes in octave score, selec individually.

5. You can delay t value. The Del the track sta s after "Delay" a Press RETUR

6. You can kee playback mo playback. You to blank or "

7. You can exper Transpose, a l

LOADING AND

To load and pl y memory. You can until all the tra s window. (Click or delete the Tape Tape Deck icon the Track List will command to loa a

4. You can transpose the pitches of any single track by setting either the Key or the Oct sliders in the Track Control Box. They're located in the bottom half of the box. The Key slider transposes the pitch of the track's notes up by 0 to 11 half-steps. The Oct slider transposes the pitch of the track's notes in octaves up or down 4 octaves. To transpose all the tracks in a score, select each track and set the transposition for each score individually.

5. You can delay the beginning of the track's playback by changing the Delay value. The Delay value is the main counter reading where the first note in the track starts to play. To change the Delay value, click on the number after "Delay" and then use the editing keys to change it to a new number. Press RETURN when you're finished.

6. You can keep any individual track from playing by clicking on the playback mode button until "Mute" appears. This silences the track during playback. You can play the track again by setting the playback mode back to blank or "Thru".

7. You can experiment with the special operating modes Echo, Trigger, Transpose, and Match. See **Tape Deck Operating Modes.**

LOADING AND PLAYING A NEW SCORE

To load and play a new score, you first have to get rid of the score in memory. You can click on the Delete command at the top of the Track List until all the tracks are deleted, but a faster way is to close the Tape Deck window. (Click on Yes when a requester asks you if you're sure you want to delete the Tape Deck.) Re-open the Tape Deck by double-clicking on the Tape Deck icon in the Patch Panel. When the Tape Deck window reappears, the Track List will be empty. Once the Track List is empty, select the Load command to load a new score, and set the tracks in that score for playback if

necessary. If you don't clear the tracks of an old score from the Track List before you load a new score, the new score tracks will simply be added to the old score tracks in the Track List. Who knows, you could stumble on something beautiful. Or your music might end up sounding like stereo wars in a college dormitory.

§§§§§§§§§§§§§§§§

R

Recording sc e
SoundScape. Lik
Deck to lay do n
at time. The Tape
multi-channel tap
recording not a ai

In this chapter 'c
own recording.
record using the T
In and Punch (it
and will learn how
a time.

MANAGING

To create your (vi
how to create, del
notes and other M
you must create a
You can create as i

tl.. Track List
be added to the
lc stumble on
ike stereo wars

Recording Scores

Recording scores is one of the most enjoyable and creative uses of SoundScape. Like a musician in a recording studio, you can use the Tape Deck to lay down track after track of music, building up compositions a layer at time. The Tape Deck gives you the flexibility of a full mixing panel and multi-channel tape deck, and gives you additional features like step time recording not available on any standard tape deck.

In this chapter you'll learn how to handle Tape Deck tracks to make your own recordings. You'll see how to set up tracks for recording, and how to record using the Tape Deck buttons. You'll also learn how to use the Punch In and Punch Out registers to record over specific sections of your tracks, and will learn how to use step time recording to make recordings one note at a time.

MANAGING TRACKS

To create your own scores, you must know how to manage Tape Deck tracks; how to create, delete, duplicate, and move tracks. Since each track records notes and other MIDI data from any one of the SoundScape input modules, you must create a track for each MIDI message source you want to record. You can create as many tracks as memory will hold.

You use the six commands at the top of the Track List in the Tape deck to manage tracks. The six commands and their functions are:

- The **Create** command, which creates a new track in the Track List.
- The **Dupe** command, which makes a duplicate of any track you have selected in the Track List. (You select a track by clicking on it.) The duplicate track includes all the contents of the track and also all the track's settings in the Track Control Box.
- The **Top** command, which moves the selected track to the top of the Track List.
- The **Delete** command, which deletes the track you have selected in the Track List.
- The **Save** command, which opens a requester where you can save all the tracks in the Track List as a score.
- The **Load** command, which opens a load requester where you can choose and load a saved score from a disk. When you load a score, SoundScape adds all its tracks to the tracks already in the Track List.

PREPARING TRACKS FOR RECORDING

To prepare a new score for recording, you might wish to close and reopen the Tape Deck window to delete any tracks already in the Track List. (Make sure to save them as a score first if you don't want to lose them for good!) To create a single new track and then prepare it for recording, follow these steps:

1. Select the Create command above the Track List. A new track will appear at the top of the Track List.

2. You can cha ge above the Tracl Amiga's edit g To thorough.y name and pres to 12 charact s

3. Click on the op Control Box n

4. The icon just e this track. o Keyboard, alth the icon to c l Keyboard icon automatically c

5. Just below the number you t MIDI data on. through chan e channel that the in the control f from the inpu n set to send data example.

The track is no v channel 1. If y u can send the incor same time as it' b the Sampled Soun to set up the trac u

2. You can change the name of the track by clicking on the name that appears above the Track Control Box. A cursor will appear, and you can use the Amiga's editing keys to replace "No Name" with a name of your choosing. To thoroughly clear the old name, put the cursor at the beginning of the name and press the DEL key 12 times. You can then type in any name of up to 12 characters. Press RETURN when you're done.

3. Click on the operating mode button in the upper left corner of the Track Control Box until it reads "Record".

4. The icon just below the operating mode button shows the input module for this track. For this example, you can use the convenient Console Keyboard, although other input modules will work as well. Just click on the icon to cycle it through the available input modules until the Console Keyboard icon appears. When you select an input module, SoundScape automatically connects it to the Tape Deck in the Patch Panel.

5. Just below the input module icon is the input MIDI channel number. The number you set here determines the MIDI channel that this track receives MIDI data on. You can change the number by clicking on it to cycle through channel numbers. The input MIDI channel must be set to the same channel that the input module uses as an output channel (which you can set in the controls for that specific module), or else the MIDI data won't make it from the input module to the track. Since the Console Keyboard is usually set to send data on channel 1, set the input MIDI channel to channel 1 for this example.

The track is now set up to record from the Console Keyboard on MIDI channel 1. If you want to monitor the MIDI data coming into the track, you can send the incoming MIDI messages out to the track's output module at the same time as it's being recorded. Although you can use any output module, the Sampled Sound Player is convenient for this example. Follow these steps to set up the track to play out to the Sampled Sound Player while it records.

1. Set the output mode button in the upper left of the Track Control Box to "Thru" by clicking on it until "Thru" appears. (A blank or "Mute" setting won't pass on the incoming signals to the output module.

2. Set the output module to the Sampled Sound Player.

3. Set the output MIDI channel to 1.

4. Make sure there is an instrument loaded in channel 1 of the Sampled Sound Player.

To test the results, try playing a few notes on the console keyboard. Since the track passes along incoming data whether the Tape Deck is running or not, you should hear your notes being played by the Sampled Sound Player. If you don't hear notes, try selecting the Console Keyboard window to make sure the Console Keyboard module is sending out MIDI note messages.

This example sets up just one track for recording. You can follow the same steps to set up more tracks, recording from different input modules, or from the same input module. However, if you have two tracks set to receive from the same input module and MIDI channel, only one, the higher track in the display, will get them. This has important repercussions. Typically, when recording a song, you will use one device, say the console keyboard, as the input module for all the tracks. After recording on track 1, you attempt to record on track 2, but nothing happens. This is because track 1, though it is not set to record any longer, still grabs the note events because it is higher up in the list of tracks. One solution is to set track 1 to listen to some other input module or MIDI channel. A better solution is to simply put track 2 at the top (use the **TOP** command), ensuring that it gets first shot at the note events as they come in.

USING TI E RECORD

Once you've creat
Tape Deck cor c
from the begin n

1. Make sure th (

2. Click on the R
that won't let h
on.

3. Click on the P
Deck will reco
reading "000 0

4. Play notes on t
connected to t
Stop button to s

5. Click on the
you just record

To play back wha
buttons. The sco
monitors in the r

RECORDIN

If you record a sc
want at its end y
Follow these ste s

USING THE TAPE DECK CONTROLS TO RECORD

Once you've created and set up the tracks you want to record, you can use the Tape Deck controls to actually start recording. Follow these steps to record from the beginning of the tracks you set up in Record mode:

1. Make sure the Clock is turned on.

2. Click on the R button to turn it on. It will turn red. This is a safety button that won't let the Tape Deck record on any track unless the button is turned on.

3. Click on the Play From Beginning button to start recording. The Tape Deck will record on all tracks set to Record mode, starting at counter reading "000000".

4. Play notes on the Console Keyboard (or whatever input module you have connected to the tracks). When you're finished recording, click on the Stop button to stop recording.

5. Click on the R button to turn it off. This makes it safe to play back what you just recorded without worrying about recording over it.

To play back what you just recorded, use the Play or Play From Beginning buttons. The score will play back through the output modules you set as monitors in the Track Control Box.

RECORDING OVER SECTIONS OF THE SCORE

If you record a score that sounds good at its beginning, but isn't what you want at its end, you can record over the MIDI data at the end of the score. Follow these steps:

1. Set only the tracks you want to record over to Record mode. Set the other tracks to Play mode.

2. Use the Play, Fast Forward, or Rewind button to advance the counter to the beginning of the section you want to record over.

3. Stop the Tape Deck.

4. Turn on the R button.

5. Click on the Play button to start recording.

6. Click on the Stop button when you're finished recording.

7. Turn the R button off for safety.

When you play back the score, you'll hear the track you set to recorded over play back as it was originally up to the point where you started recording. You'll hear the new notes you recorded from that point on.

USING THE PUNCH IN AND PUNCH OUT REGISTERS

If you want to record over a section in the middle or end of your score, and you want to be able to listen to the music up to the point where you want to start recording, you can use the Punch In and Punch Out registers to turn the R button on and off for you. These two registers are in the upper right corner of the window. The Punch In register is labelled "PI", the Punch Out register is labelled "PO". The Punch In register stores the counter location where the R button will turn on, the Punch Out register stores the counter location where the R button will turn off. Follow these steps to use them:

1. Play your score back from the beginning.

2. When you h a
click on the "C
exact readin c

3. When you hea
click on the C
the exact readi

4. Make sure t e
the other track

5. Turn the R but

6. Start playin b
the reading in

7. When the c
turn on and the
section.

8. When the cou
will turn off n

When you play b
new section, y
recording over t
Punch Out cou t
use them to keep
want. To get r
Tape Deck's m
000000. If you d
which would n

2. When you hear the score get to the spot where you want to start recording, click on the "C" button next to the PI register. The register will capture the exact reading of the counter when you click.

3. When you hear the score get to the spot where you want to stop recording, click on the "C" button next to the PO register. The register will capture the exact reading of the counter when you click.

4. Make sure the tracks you want to record over are set to Record mode and the other tracks are set to Play mode.

5. Turn the R button off if it isn't already.

6. Start playing back the score from a counter location that starts well before the reading in the Punch In register.

7. When the counter reaches the Punch In register location, the R button will turn on and the Tape Deck will start recording. Play what you want in this section.

8. When the counter reaches the Punch Out register location, the R button will turn off and the Tape Deck will stop recording.

When you play back the score after using Punch In and Punch Out to record a new section, you should hear the new notes you played inserted in the score, recording over the notes that were there before. Since the Punch In and Punch Out counter readings stay the same unless you change them, you can use them to keep recording over the same section until you get the results you want. To get rid of Punch In and Punch Out, use the Rewind button to set the Tape Deck's main counter to 000000, then set both the PI and PO registers to 000000. If you don't do this, you could have an accidental Punch In later on, which would not be appreciated (but UNDO can save you.)

Punch in and out can also be controlled directly by clicking on the R button as the tape deck is playing. Click once, the red light turns on and you have punched in. Click again, the light turns off - punch out. If the mouse is a bit too cumbersome, you can also use the Console keyboard. If the Console Keyboard is hooked up as an input to the Tape Deck, the 'Enter' key will toggle punch in and punch out. What is particularily powerful about using the mouse or the keyboard for punching in and out is you can do multiple punches in one take, just as would be true on a mechanical tape deck.

USING MIDI FILTERS

If you record performances on an external synthesizer with controls like a pitch bend wheel or a keyboard with aftertouch, it's possible to use up memory very quickly; a few seconds of pitch wheel use can generate hundreds of separate MIDI messages to be stored in a track. If you want to filter out events like these to save memory, you can use the MIDI filters, located in the middle of the Track Control Box.

There are two separate MIDI filters: an input filter, labelled "St In" (for Status In), and the output filter, labelled "St Out" (for Status Out). If you click on either of these two labels, a window will open listing the different MIDI events you can filter. The events listed here are the same as the events stored in the Event List of the Edit Sequence window, and are explained in the next chapter. At the end of each event name is a space that shows a check mark if that event can pass through the filter or a blank if the event won't pass through the filter. To change the event status, just click on the event name to alternate between the check and the blank.

Setting events for the input filter filters MIDI data coming in through the track's input module. Setting events for the output filter filters playing back from the track to the track's output module. Filtering out events with the

input filter can save memory in a track as you record. Filtering out events with the output filter can change a track's playback without actually changing the track's contents. For example, if you have a sequence including program changes that change a synthesizer's presets, you can filter out the program changes so the sequence will play back without preset changes.

STEP TIME RECORDING

When you use the Clock with the Tape Deck to record a performance, the Clock provides a steady stream of timing clocks that the Tape Deck uses to record your performance just as it happens. If you happen to play a rhythm sloppily or hit a wrong note, the Clock keeps ticking away and the Tape Deck records it all, bad notes and good notes alike. If you're less than a sterling performer, or you are a sterling performer and want to record some impossible licks on the Tape Deck, then you can use the *step time* recording technique to freeze time in chunks as you record so you can enter notes one at a time.

The module that makes step time recording possible is the Console Keyboard. You'll recall from the first chapter of this section that the Console Keyboard can send out MIDI clocks as well as notes. If you press any of function keys 1 to 5 or the space bar on the Amiga's keyboard, they will send out a set chunk of MIDI clocks. You can connect the Console Keyboard to the Tape Deck in the Patch Panel so that whenever the Console Keyboard sends out a number of MIDI clocks, they register on the Tape Deck's main counter.

To record using the step time recording technique, you use the MIDI clocks from the Console Keyboard instead of the Clock as you record. To see how it works, try an example by following these steps:

1. Create a new, blank track on the Tape Deck.

2. Set the new track's operating mode to Record, using the Console Keyboard as the input module on MIDI channel 1. (You can use other input modules for step time recording, but for this example, the Console Keyboard is convenient.)

3. Set the track's playback mode to Thru, using the Sampled Sound Player as the output module on MIDI channel 1.

4. Make sure there's an instrument loaded in channel 1 of the Sampled Sound Player. If you play notes on the Amiga's keyboard now, you should hear sound from the Sampled Sound Player if the track is set up properly.

5. Make sure the Clock is turned off so the Tape Deck won't get any MIDI clocks from the Clock. If it isn't off, click on the Stop button in the Clock window.

6. Turn on the R button in the Tape Deck window so you can record on the new track.

7. To start step time recording, click on the Play From Beginning button in the Tape Deck window. This resets the main counter and sets the Tape Deck up to receive MIDI clocks from any input source.

8. Hold down the Q key on the keyboard to play a note pitched at C. Since the Tape Deck is recording and its main counter reads "000000", it records the beginning of this note at 0.

9. While you hold down the Q key, press and release the F5 key to send 6 MIDI clocks to the Tape Deck. The Tape Deck advances its recording time to 6, even though the main counter doesn't show it. (The main counter only updates its display every 30 MIDI clocks.)

10. Release th
C you jus
recordi g.

11. Repeat he
note pit_ne

12. Repeat ie
U, and r.
lasting e a

13. Click on th

14. Turn of h

To play bac y
then use the J
You should he

As you use ste

You must a a
Deck in the P
Keyboard t th
doesn't matte

Even thoug yo
to start, so you
time recordi g.

Be sure the Clo
the Clock or y

As you record
want, press t

10. Release the Q key. The Tape Deck records the end of the note at 6, so the C you just played is recorded as stretching from 0 to 6 clocks in the recording.

11. Repeat the last three steps with the W key instead of the Q key to record a note pitched at D that stretches from 6 to 12 clocks in the recording.

12. Repeat the same step time recording technique using the keys E, R, T, Y, U, and I. This finishes recording a C major scale of 8 notes, each note lasting exactly 6 MIDI clocks long.

13. Click on the Stop button to stop recording.

14. Turn off the R button for safety.

To play back your score, make sure the Clock is attached to the Tape Deck, then use the Go From Beginning button to play back what you just recorded. You should hear a perfectly even C major scale.

As you use step time recording, keep this things in mind:

You must always have the Console Keyboard module connected to the Tape Deck in the Patch Panel for the MIDI clocks to pass from the Console Keyboard to the Tape Deck. Since clocks don't use any MIDI channel, it doesn't matter what channel you set the Console Keyboard to.

Even though you are step time recording, the Tape Deck still needs to be told to start, so you must always hit the Play or Play From Beginning to start step time recording. To stop recording, hit the Stop button.

Be sure the Clock isn't sending clocks to the Tape Deck by either turning off the Clock or by disconnecting it from the Tape Deck in the Patch Panel.

As you record notes using step time, hold down the key with the pitch you want, press the function key or space bar you want to give the note the length

you want, then release the pitch key. If you use the standard MIDI equivalency of 24 MIDI clocks to the quarter note, then the Console Keyboard as it's originally set up sends out these note lengths:

Space Bar: an eighth note (12 MIDI clocks)

F5: a triplet eighth note (8 MIDI clocks)

F4: a sixteenth note (6 MIDI clocks)

F3: a triplet sixteenth note (4 MIDI clocks)

F2: a thirty-second note (3 MIDI clocks)

F1: a triplet thirty-second note (2 MIDI clocks)

If you want a note to last longer than any of these lengths, just press several length keys in a row while you hold a pitch key down.

You can use the Edit Console Keyboard window to change the number of MIDI clocks each length key sends out if you want to change the function keys and space bars to create different length notes.

USING THE KEYBOARD TO CONTROL THE TAPE DECK

As you use step time recording, you'll find that it's sometimes inconvenient to take your fingers from the keyboard to use the mouse when you need to click on the Tape Deck buttons. SoundScape offers an alternative: Amiga keyboard keys that directly control the Tape Deck. These keys perform the same functions as the Tape Deck buttons:

- The right cursor key plays the Tape Deck by sending out a MIDI Start message (same as the Play button).

- The left cursor key resets the main counter to "000000" by sending out a MIDI Reset to 0 message and plays the Tape Deck by sending out a MIDI Start message (same as the Play From Beginning button).
- The down cursor key stops the Tape Deck by sending out a MIDI Stop message (same as the Stop button).
- The ENTER key turns the Record button on and off.

SAVING SCORES AND ENVIRONMENTS

When you're finished recording a score, you can save all its tracks by selecting the Save command above the Track List. A save requester that works just like the instrument save requester will appear. Type in the name you want to save the score under and then press the RETURN key to save the score. SoundScape saves the contents of all the tracks as well as the track settings.

If you set up your score to work with the Sampled Sound Player, the Console Keyboard, and other SoundScape modules, you can save the entire arrangement as an environment by selecting the Save command in the Patch Panel. This saves the score, the Sampled Sound Player's instrument bank, the Patch Panel connections, and the settings of all the other SoundScape modules. It doesn't save the sampled sound instruments you're using, so if you created some new instruments or modified some old ones, you should go to the Sampler window to save each one individually to complete saving the environment.

§§§§§§§§§§§§§§§

Editing Sequences and Songs

Each time you record music on a track, you create a *sequence* of notes and MIDI events in the Amiga's memory. You can look at and edit the contents of each sequence you create in the Edit Sequence window. Using another window called the List Store window, you can copy sequences from one track and use them in another track. In the Edit Song window, you can take sequences from the List Store window and tie them together into sequence combinations called *songs* .

To make it easy to deal with editing sequences and songs, you can use a helpful analogy; think of each track in the Tape Deck as a movie. Each sequence you record in a track is like a strip of film containing a series of events in a scene: hero slugs villain, villain trips hero, hero activates helicopter hat and flies away, etc. In a track sequence, the events are more practical events like notes, pitch wheel readings, and other musical data. When you record a sequence, the timing clocks that the Tape Deck receives are like film frames; just as a movie event can start at frame 46 and last until frame 59, a sequence event can start at timing clock 32 and last until timing clock 77. When you create a song by tying together sequences, it's much like splicing together the film scenes on strips of film to create a full movie on a reel. You'll see more applications of this analogy as you learn more of SoundScape's editing capabilities.

EDITING A SEQUENCE

To examine and edit a sequence in a track, first select the track you want in the Track List in the Tape Deck window. Then select the Sequence command at the bottom of the Track Control Box to open the Edit Sequence window. When it opens, you'll see that it has several sections:

- The **Event List** is the box in the right of the window. It lists the events in the sequence in order from top to bottom. If there are more events than the box can display, you can drag the scroll bar to the right of the list up and down to see different parts of the sequence.

- The **event commands** are the commands in black letters in the upper left of the window. You can use these commands to manipulate the events in the Event List.

- The **event data** is the section of words and numbers in white in the lower left of the window. This section shows you information for whichever single event you select in the Event List.

USING THE EVENT LIST

The events in the Event List are the events recorded in the track the last time you recorded using the Tape Deck. Each event shows information in this order:

- The MIDI clock number on the Tape Deck's main counter when the event occured.

- The type of event

- One data value for the event.

- A second data value for the event (optional for some types of events).
- A third data value for the event (optional for some types of events).

You can select any single event in the Event List by clicking on it. SoundScape will highlight the event and show all the data for the event in the event data section of the window.

Each event in the Event List has a pointer called the Next Pointer that points to the next event in the sequence. You can see the Next Pointer for the selected event in the Next Pointer display (the box below the Event List), where it displays the next event to follow the selected event. The last event in the sequence normally has no Next Pointer because there is no next event.

Once you have selected an event, you can use the event commands to work with it by selecting the command you want:

- **Create** creates a new event that is a copy of the selected event and places it just below the selected event in the Event List.
- **Delete** erases the selected event.
- **Save** opens a requester where you can save the entire contents of the Event List to disk as a sequence.
- **Load** opens a requester where you can load any sequences that were previously saved to disk.
- **Copy** copies the top sequence referred to in the List Store window into the Event List, replacing the sequence already in the Event List *or* when the Next Pointer display is selected, adds a complete copy of the top sequence referred to in the List Store window to the Event List without totally replacing the existing sequence. (The List Store, Next Pointer display, Copy, and the Use command that follows are all discussed at length later, so don't worry too much about them now.)

• **Use** erases the sequence in the Event List and refers the Event List to the top sequence in the List Store window *or* when the Next Pointer display is selected, refers the Event List from the selected event to the top sequence referred to in the List Store window.

• **Q'tize** (short for Quantize) changes the MIDI clock locations of all the events in the list to fit on multiples of whatever value follows the Q'tize command. Change the quantize value by clicking on it and editing it with a cursor. You can use Q'tize to clean up sloppy rhythms in a sequence. Once you use Q'tize, you can't get back the original rhythms, so you should make a backup of the track by using Dupe in the Track List before you open up the Edit Sequence window and use the Q'tize command. If you don't like the new quantized version of the track, you can always delete the track and use the duplicate track again.

CHANGING THE EVENT DATA

You can change a selected event by changing its data in the event data section of the window. To change the data, you just select the number you want to change, use the Amiga's editing keys to change it, then press RETURN when you're finished.

The data for each event always has a Time value at the top of the event data. This is the MIDI clock reading where the beginning of the event was recorded. At the bottom of the event data is a Next value that is the number of MIDI clocks that elapse from the beginning of this event until the start of the next event. By changing the event's Time value, you can change its location in the sequence. By changing the event's Next value, you can move the events following the selected event closer or further from the beginning of the selected event.

The event data that the Edit Sequence window displays between the Time and Next values varies depending on the kind of event that is selected in the Event

List. The second data line is a white box with black lettering. This box shows the event type. By clicking on the box, you can cycle the event through all the different event types. As you change types, the data lines also change. The following list shows the different event types and the data displayed for each event type (not including Time and Next, which are common to all event types):

• **Note On/Off**, shown in the Event List as:

(starting time) **Nt** (letter pitch) (octave number) (note length)

Note On/Off records the beginning and ending of a note, and also records the velocity used to play the beginning of the note. The data for Note On/Off is:

Note: a value from 0 to 127 for the pitch that was recorded. 60 is middle C (C4). Raising or lowering the value by one raises or lowers the pitch by a half-step.

Velocity: a value from 0 to 127 for the velocity of the keystroke that played the note. The higher the value, the higher the velocity.

Length: the length in MIDI clocks from the beginning of the note to the end of the note (a value anywhere from 1 to 65535).

Poly Pressure, shown in the Event List as:

(starting time) **Pp** (note) (pressure)

Poly Pressure records changing key pressure (often called *aftertouch*) for any key on a keyboard with aftertouch. The data for Poly Pressure is:

Note: A value from 0 to 127 that tells which key on the keyboard is sending aftertouch information. 60 is middle C. The higher the value, the higher the key on the keyboard.

Pressure: A value from 0 to 127 that tells the key pressure reading when this event was recorded. The higher the value, the higher the pressure.

- **Control Change**, shown in the Event List as:

  (starting time) **Cc** (number) (value)

  Control Change records a command to send to an attached synthesizer that changes the setting of one of its controls (like a modulation wheel or a volume slider, for example) to a new setting. The data for Control Change is:

  Number: The number of the control (from 0 to 127, determined by the specific MIDI synthesizer you have attached to the Amiga).

  Value: The controlling value (from 0 to 127) you want to send to the attached synthesizer to change a control setting. This value varies from control to control; its results depend on the particular synthesizer and control that you send it to.

- **Program Change**, shown in the Event List as:

  (starting time) **Pc** (patch)

  Program Change records a patch number to send to an attached synthesizer with changeable voices (patches). The patch number chooses a new patch so the synthesizer will play notes following with a new voice. The data for Program Change is:

Patch: a value from 0 to 127 that selects a new patch on an attached synthesizer. The results depend on the patches belonging to the synthesizer.

- **Mono Pressure**, shown in the Event List as:

  (starting time) **Mp** (pressure)

  Mono Pressure records the overall keyboard pressure on the MIDI channel this sequence is recording. The data for Mono Pressure is:

  Pressure: a value from 0 to 127 that tells the overall keyboard pressure. The higher the value, the higher the pressure.

- **Pitch Wheel**, shown in the Event List as:

  (starting time) **Pw** (bend)

  Pitch Wheel records the amount of pitch bend applied to the notes being played in the sequence. The data for Pitch Wheel is:

  Bend: a value from 0 to 16384 that records how much pitch bend to apply to notes. A value of 8192 is no pitch bend at all; anything above that bends pitch up, anything below that bends pitch down.

You can fine tune any event you care to by selecting it in the Event List and changing its data. If you want to add new events, you can use the Create command to create an event, then select the type of event you want with the qualities you want in the event data area. If you want to, you can create your own sequences from scratch in the Edit Sequence window this way without recording from a keyboard. (If you add events like Program Change and Pitch Wheel to your track and want to play it back, be sure the filters in the Status Out window are set to let the events pass through, or you won't hear any of their results.)

USING THE LIST STORE

Think of the List Store as a row of five film clips that each store a film strip by clipping onto its first event. The List Store window, which opens as soon as you select an event in the Edit Sequence window, has five slots that act like film clips: each one stores a sequence of events by holding its first event in the slot. This first event stored in the slot is called a *sequence referral* , because it isn't actually the sequence itself. The rest of the sequence is stored elsewhere in the Amiga's memory; the event stored in the slot just refers to the sequence.

STORING A SEQUENCE IN THE LIST STORE

Double-click on any event in the Event List to store it in the List Store. SoundScape will store all the events in the sequence--from the event you double-clicked to the event at the end of the sequence--in the Amiga's memory, and then put the first event of the sequence in the List Store's top slot (called the *active slot*). Any events in the Event List before the event you double-clicked won't be stored. The top slot of the List Store window will show the first event of the sequence you just stored, preceded by the prefix "Seq:", which stands for Sequence.

You can use the List Store's active slot to copy a sequence into other tracks or into the Edit Song window. The other four slots in the List Store, called *storage slots* and located beneath the active slot, let you store additional sequence referrals. To copy a sequence referral from the active slot to a storage slot, follow these steps:

1. Put the pointer on the top slot and hold down the left button on the mouse.

2. Drag the pointer to the storage slot you want to copy to (a slot outline will follow the pointer down to the storage slot.)

3. Release the mouse button. The sequence referral will be copied into the storage slot, replacing any sequence that might already be there.

To return a sequence referral from a storage slot to the active slot, double-click on the storage slot you want. The contents of the slot will be copied to the top slot, replacing any sequence in the top slot.

CLEARING SEQUENCE REFERRALS FROM THE LIST STORE

Clearing sequence referrals from the List Store is quite easy. Next to the top slot is a small white button with an "E" in it. The E stands for empty. If you double-click on the E button, the top slot will be cleared out. You can then copy the empty top slot to any of the other slots in the List Store you want to empty.

One word of caution: sequences that are referred to in the List Store are held in the Amiga's memory by that sequence referral just like a strip of film is held in the air by a film clip. If you clear a sequence referral from a slot and the sequence doesn't exist in the Edit Sequence window's Event List or in the Edit Song window, the sequence will be erased from the Amiga's memory much like releasing a strip of film from a film clip will let the film fall to the cutting room floor, never to be seen again.

COPYING THE CONTENTS OF THE ACTIVE SLOT TO THE EDIT SEQUENCE WINDOW

You can copy the the sequence referred to in the List Store's top slot to the Edit Sequence window by first selecting an event in the Event List and then using the Copy command. For example, to copy the top slot sequence to the Edit Sequence window, follow these steps:

1. Select any event in the Event List.

2. Select the Copy command. The sequence in the top List Store slot will replace the entire sequence in the Event List.

If you use the Use command instead of the Copy command in the same circumstances, the results will look the same, but there is a difference: The Copy command actually copies all the events in the List Store's top sequence and puts them in the Event List. The Use command just copies the top sequence's referral to the Event List and doesn't actually copy all the events in the sequence.

The practical difference between the two commands is that each time you use the Copy command, you use up more of the Amiga's memory, because Copy creates new events to be stored, while each time you use the Use command, you use up very little memory, since no new events are created; they're just referred to. Another practical difference is that any time you change the original sequence that you put into other tracks and sequences with the Use command, the events in the other tracks and sequences will change with the original sequence. If you put the same sequence into other tracks and sequences with the Copy command, the copied sequences won't change at all when you change the original sequence, since the copies are stored separately in their own memory.

CHANGING THE EVENT LIST POINTER WITH COPY AND USE COMMANDS

By selecting the Next Pointer in the Next Pointer display, you can use Copy and Use to add a sequence to the sequence already in the Event List without replacing the entire Event List sequence. If you first select an event in the Event List and then select its Next Pointer in the Next Pointer display, the Copy and Use commands will add all the events in the top sequence in the List Store to the events in the Event List. The Next Pointer display will change to show the first event in the top slot sequence. The entire top slot sequence will replace any events in the Event List following the selected event.

The difference between Use and Copy in this situation is the same as before: Copy actually copies all the events in the List Store's top sequence, Use just points the Event List to the List Store's top sequence.

CREATING AN ENDLESS LOOP

One very handy use of the Use command is to create an endless loop by replacing a selected event's Next Pointer with an event that occurs before the selected event. For example, follow these steps to create an endless loop that plays through an entire sequence over and over:

1. Double-click on the first event of the sequence in the Event List to store the entire sequence in the top slot of the List Store.

2. Select the last event in the Event List.

3. Select the Next Pointer display.

4. Select the Use command. The Next Pointer will change to the first event in the top slot sequence, and any empty part of the Event List will be filled with loops of the top slot sequence.

5. Change the Next value of the last event to set the loop duration. For example, if the last event occurs at time 900 and the loop should be 1000 MIDI clocks long, set Next to 100. This effectively says, "Wait 100 clicks before starting over."

If you close the Edit Sequence window and play the track on the Tape Deck, you'll hear the sequence played over and over.

You can also create loops by pointing the last event's Next Pointer to a point midway in the Event List sequence, or to a point anywhere in the sequence before the last sequence.

CREATING A SONG

Once you've stored several sequences in the List Store, you can use the Edit Song window to tie them together into a song. To open the Edit Song window, first make sure the Edit Sequence window is closed, then select the Song command from the bottom of the Track Control Box in the Tape Deck window. When the Edit Song window opens, you'll see that it has several sections similar to the Edit Sequence window:

- The Sequence List window is the box in the right of the window. It lists the sequences in a song in order from top to bottom. If there are more sequence listings than the box can display, you can drag the scroll bar to the right of the list up and down to see different parts of the song.

- The sequence commands are the commands in black letters in the upper left of the window. You can use these commands to manipulate the sequence listings in the Sequence List.

- The sequence data is the section of words and numbers in white in the lower left of the window. This section shows you information for whichever single sequence listing you select in the Sequence List.

USING THE SEQUENCE COMMANDS

When you first open the Edit Song window, the Sequence List will be empty. You can't create a song by recording the way you can create a sequence. Instead, you must create the song from scratch in this window by using the Sequence commands to tie together sequences:

- **Create** copies the selected sequence listing in the Sequence List and puts the copy just below the selected sequence listing in the Sequence List. If there is no sequence listing in the Sequence List, Create makes a blank sequence listing.

- **Delete** erases the selected sequence listing.

- **Save** opens a requester where you can save the entire contents of the Sequence List to disk as a song.

- **Load** opens a requester where you can load any songs that were previously saved to disk.

- **Copy** works just like the Copy command in the Edit Sequence window, but replaces an entire song in the Sequence List, or adds sequences to existing sequences in the Sequence List if you select the Next Pointer, or selects the actual event list if you select the Sequence Pointer.

- **Use** is identical to Copy except instead of making a copy of the list, the same list is used.

To create a song, you first create some sequence listings using the Create command. Once you have sequence listings in the Sequence List, you can read information in each listing in this order:

- The transposition status

- The first event in the sequence used in the listing.

When you first create a sequence listing, there is no sequence loaded in the listing, and no set transposition. You can change the sequence data for the listing to load a sequence and set its transposition. There are four different data settings in the sequence data area:

- **Sequence** is the third line in the data area. It's just the title for the listing in black letters just below it called the sequence content box. The sequence content box shows the first event of the sequence in the sequence listing, and works like the slots in the List Store window: it's a sequence referral that hangs on to a sequence in the Amiga's

memory. (The sequence content box is probably blank now, referring to no sequence at all, unless you've created sequence listings in the Sequence List.) To change the sequence, or to load a sequence into an empty sequence listing, select the black letters. They'll be highlighted. Select the Use command to load the sequence from the top slot of the List Store to the sequence listing. If you want to see the contents of the sequence you just loaded, double-click the black letters to open the Edit Sequence window. You can see the contents of the sequence in the Edit Sequence window's Event List.

• **No Trans/Transpose** is the white box with black lettering at the top of the sequence data area. Click on the box to change it between No Trans and Trans. When the sequence listing is set to Trans, all the notes in the sequence play back transposed up or down by the value set in the Trans listing. When the sequence listing is set to No Trans, all the notes in the sequence play back at whatever Trans value was set in the previous sequence listing. If there are no previous sequence listings with a Trans setting, then a No Trans setting plays back all the notes in the sequence at their original pitches.

• **Trans**, just below No Trans/Transpose, lists the number of half-steps the notes in the sequence will be played above or below their original pitches. You can change the value by clicking on the number after Trans and using the editing keys, pressing RETURN when you're done. Negative numbers transpose down from the original pitch, positive number transpose up from the original pitch.

• **Wait**, at the bottom of the data area, shows the number of MIDI clocks that SoundScape will wait after playing the preceding sequence before playing this sequence. By changing this value, you change the spacing between sequences. You can change it by selecting the value and using the edit keys, then pressing RETURN when you're finished.

USING THE LIST STORE WITH THE EDIT SONG WINDOW

You can store songs in the List Store much like storing sequences from the Edit Sequence window: double-click on a single sequence listing in the Sequence List. SoundScape stores all the sequence listings from the selected listing to listing at the end of the song in the Amiga's memory and puts a song referral in the top slot of the List Store. (Sequence listings before the selected listing aren't stored.) The List Store shows song referrals by listing the first sequence listing in the song. Song referrals stored in the List Store have the prefix "Song:" before them, while sequence referrals stored from the Edit Sequence window have the prefix "Seq:".

Once you save a song to the List Store, you can change the song in the Sequence List and the song Next Pointer just below the Sequence List the same way you change sequences and sequence Next Pointers in the Edit Sequence window using the Copy and Use commands. The commands work the same way here, but copy or refer to songs and sequences in songs instead of to sequences and events in sequences. You can even create loops if you want to.

Since you can store both sequences and songs in the List Store, you may at some time try to insert a sequence where you should put a song, or a song where you should put a sequence. SoundScape will catch the mistake, and won't let you put the wrong kind of element in an Edit window.

PLAYING A SONG

When you've finished creating a song by tying together sequences, you play the song by closing the Edit Song window and using the Tape Deck just as you would to play back a sequence. It plays back on just one track, the track you created it in, and can be controlled with the clock speed, track transposition, and other track controls. Since the track you used no longer contains a sequence, but a linked set of sequences, SoundScape will protect it from

recording by making it impossible to record on the track, even if you set it to Record mode and turn on the Tape Deck's R button.

§§§§§§§§§§§§§§§§

Tape Deck Operating Modes

When you used the tape deck to play back and record scores in previous chapters, you used the Track Control Box to set the operating and playback modes for different tracks. As you cycled through the operating modes to get from Record to Play, you may have noticed other modes: Echo, Match, Trig, and Trans. These modes make the Tape Deck run in different ways: the Echo and Trig modes can let you play riffs at the press of a single key; the Match mode can turn the Tape Deck into an accompaniment machine, following your tempos as you play on a keyboard; the Trans mode can let you instantly transpose selected tracks to different keys as you play back a score. In this chapter, you'll learn how to use each of these modes.

ECHO MODE

To use a track in Echo mode, you must first record a sequence in the track. When you set the track to Echo mode and then then press either the Play or Play From Beginning button on the Tape Deck with the Clock turned on, the Tape Deck waits for a note from the track's input module (like the MIDI In module or the Console Keyboard module). As soon as it receives a note, it checks the distance of the pitch of the note from the C in the fourth octave (C4, which is middle C). It then starts to play the track's sequence transposed

up or down by the same distance the incoming note is from C4. For example, if you have an arpeggiated C flat seven chord recorded in the track, it will play back as a F flat seven chord if the track receives an F4 note from the input module. If you change the octave or key setting for the Echo track, it will be transposed additionally by this amount.

The Tape Deck will play back the full sequence in the track from beginning to end each time it receives a note, so when many notes come into the track at once, the Tape Deck will play many sequences at once (up to the limit of the synthesizer playing the notes).

To see how Echo mode works, try this example:

1. Select the Console Keyboard (or MIDI In) as the input module for an empty track in the Tape Deck.

2. Record a short sequence in the track. An arpeggiated flat seven chord like C - E - G - Bb - G - E will work well.

3. Set the track to play back using a Sampled Sound Player instrument.

4. Set the track's operating mode to Echo.

5. Start the Tape Deck playing by clicking on the Play button.

6. Press any playing key on the Amiga's keyboard (or synthesizer keyboard if you're using MIDI In. The sequence you recorded will play from beginning to end starting on the pitch you pressed. If you press more keys, more sequences will play starting on the pitches of the keys you press.

Ofcourse, you can trigger an Echo track off of another prerecorded track by routing through the Mixer. With several Echo tracks and corresponding control tracks you can actually piece together a song. For example, track a is one theme, track b is the control track that determines when that theme comes in and at what transposition. Track c is another theme and track d controls

that. You can play with multiple copies of each staggered in time and transposed wild amounts, or maybe put together a conventional AABA format song.

TRIGGER MODE

Trigger mode is very similar to Echo mode in that you can use it to make a sequence play automatically. Unlike Echo mode, Trigger mode doesn't transpose the sequence, multiple triggers simply reset and restart the sequence (as opposed to playing yet another copy of it), and a specific event, not just any note, is required to trigger the sequence.

Because we need to know what event to trigger on, the first event in the sequence is reserved as a template to tell the Tape Deck what to look for. This event will not be played.

To see how Trigger mode works, try this example:

1. Create a new track, put it at the Top, and select the Console Keyboard as the input module for it.

2. Record a short sequence in the track, but have the first note be C ('Q').

3. Set the track to play back using a Sampled Sound Player instrument.

4. Set the track's operating mode to Trig.

5. Start the Tape Deck playing by clicking on the Play button.

6. Press any playing key but 'Q' on the Amiga's keyboard. Nothing happens. Press 'Q'. Your sequence is played, but starting with the second note. If you press 'Q' again while it is still playing, it will reset and start again.

If you kept the Echo track from the previous example, you probably noticed that whenever the 'Q' key was pressed, the Trigger track played, whereas all other note events set the Echo sequence running (assuming the Console Keyboard was the input module for that track.) The Trigger track will only listen to the specific event intended for it. All others are passed on. This has some nice advantages. You can create a set of different sequences, each triggered by a different note, but all on the same MIDI channel and port. Then, use one keyboard to to set them all off.

MATCH MODE

Like Echo and Trigger modes, Match mode needs a sequence recorded in a track before it will work. When you set a track to Match mode and press the Play or Play From Beginning button, the Tape Deck plays the first note in the track and waits for the same note to be played. If the note is already held down, it happily continues. If not, it stops just before it would release the note and waits for the performer to get it right. It continues doing this with each note, starting and stopping as it runs through the piece.

Match mode is useful for learning scales, riffs, and songs on an attached MIDI keyboard. For example, you can first record a scale in a track, and then set the track to Match mode. You can select the MIDI In module (bringing signals from the MIDI keyboard) as the track's input device and the Player Piano (discussed in the next section) as the track's output device. When you press the Play button, the Player Piano window displays the first note in the scale on its keyboard. If you play the same note on the MIDI keyboard, the Tape Deck will play the next note in the scale which is displayed on the Player Piano window, and will wait for you to match that on the MIDI keyboard. If you connect the MIDI keyboard directly to the Player Piano using a different MIDI channel, you can see the scale in the track and what you actually play displayed in different colors on the Player Piano keyboard.

TRANSPOSE MODE

Transpose mode lets you change the key of a score playing back on a Tape Deck as it plays. To use it, you first enable every track in the score you want to transpose, then you create a new track as a transpose control track that receives notes from an input module. When you play back a score, the Tape Deck monitors the notes coming into the transpose control track, and transposes all the tracks you enabled earlier to match the pitches coming in. To see how Transpose works, try this example:

1. Load any score from the SoundScape disk using the Load command in the Tape Deck window.

2. Select the first track in the Track List.

3. In the upper right corner of the Track Control Box is a small white box. Click on the box to make a T appear. You've now enabled this track to respond to the Transpose track.

4. Select each of the rest of the tracks in the Track List and set the "T" button to T so they're enabled to respond to the Transpose track too.

5. Create a new track.

6. Set the input module of the new track to the Console Keyboard.

7. Set the operating mode of the new track to Transpose.

8. Make sure the Record button is off on the Tape Deck, then click on the Play From Beginning button to start the score playing.

9. As the score plays, press any playing key on the Amiga's keyboard. The notes in the other tracks set to T will from that point on be transposed by

the key
change the t

Like Echo
notes in the t
playback
transpose
button by
tracks set
the T tracks w

You should
receiving
sequence

Transpose
want to
12 bar blues
play over
another track
change the
to use with a
when you're
In module
track to send
track to the
for the Transp

SEEING

As you create
many

the key you pressed. When you press more keys, each keypress will change the transposition of the sequence playback.

Like Echo mode, Transpose checks the distance of the pitch of incoming notes in the transposing control track from C4, and transposes the score playback by the same distance up or down. Remember that it will only transpose the tracks with the T button turned on. You can turn off the T button by clicking on it a second time. If you play back a score with some tracks set to "T" and other tracks not set to "T", then you can transpose just the T tracks without transposing the others.

You should only set one track in a score to Transpose mode. It serves as a receiving track only, receiving notes from its input module. If you have a sequence stored in that track, the sequence won't play back at all.

Transpose works very well if you have a looping sequence in a track that you want to change harmonically. For example, you can record just one bar of a 12 bar blues pattern, and then set it in the Edit Sequence window so it will play over and over again. You can then play back the sequence and use another track in the Transpose mode to transpose it whenever you need to change the chord. The Console Keyboard is a very convenient input module to use with a Transpose track, because its limited pitch range isn't a handicap when you're transposing, and because it frees up other modules like the MIDI In module to use for other purposes. You can also use another Tape Deck track to send in notes to the Transpose track by sending the output of the first track to the MIDI Mixer and then using the MIDI Mixer as the input module for the Transpose track.

SEEING MODE SETTINGS IN THE TRACK LIST

As you create more and more tracks in a score, you may find yourself with many different operating modes set for each track. An easy way to see the

status of each track is to look at the small square before the name of each track in the Track List. If the track's operating mode is set to Play, the square will be white. If the operating mode is Record, the square will be red, and if the mode is Echo, Match, Trigger, or Transpose, the square will be black. If you set the playback mode of the track to Mute, the box will turn to an empty square outline regardless of the track's operating mode.

§§§§§§§§§§§§§§§§

n of each
› Play, the
w l be red,
ıre will be
w ''l turn to

Additional Topics

In the rest of this manual, you've learned how to work with the Patch Panel, the Sampled Sound Player, and the Tape Deck. There are four other modules on the SoundScape disk that are easy to learn and handy to use: the Player Piano module, the MIDI Mixer module, the MIDI In module, and the MIDI Out module. You'll learn how to use them in this chapter. You'll also take a look at how SoundScape saves different files to disk, and will learn about expanding SoundScape with additional modules.

THE PLAYER PIANO MODULE

You can use the Player Piano Module as a visual check on notes coming into it over two different MIDI channels. All you need to do is use the Patch Panel to connect an output module icon (or icons) to the Player Piano icon. If the Player Piano is set up to receive MIDI messages on the channels the output modules are using to send, the Player Piano will show the notes it receives on its keyboard, flashing notes in grey that come in over one MIDI channel, and flashing notes in red that come in over a second MIDI channel. The Player Piano usually uses grey for MIDI channel 1 and red for MIDI channel 2, but you can change that (as well as incoming pitch transposition) by opening the Set Up Player Piano window.

To open the Set Up Player Piano window, double-click on the Player Piano icon in the Patch Piano. When it opens, you can use these controls:

- The **Grey Channel** slider sets the MIDI channel that the Player Piano shows with grey keys as any channel from 1 to 16.

- The **Red Channel** slider sets the MIDI channel that the Player Piano shows with red keys as any channel from 1 to 16.

- The **Octave** slider sets transposition of incoming pitches in octaves. The default is 3, where the Player Piano shows C in the 3rd octave as the bottom C on its keyboard and middle C (C in the 5th octave) in the middle of its keyboard. Setting the Octave slider up or down changes the octave span of the keyboard, shown by the the C in the bottom of the keyboard: setting the slider to 6 means the bottom key on the keyboard displays a C in the 6th octave; setting the slider down to 1 means the bottom key displays a C in the 1st octave.

- The **Key Transpose** slider sets upward transposition of incoming pitches in half-steps. The default is 0--no transposition; the maximum is 11--the Player Piano transposes all incoming pitches up by 11 half-steps.

To see how the Player Piano handles two separate channels at once, try using it on two separate tracks of a Tape Deck score:

1. Open the Tape Deck and load a score with at least two tracks.

2. Set one of the tracks to play back on MIDI channel 1, setting the Player Piano as the output module.

3. Set another track to play back on MIDI channel 2, setting the Player Piano as the output module.

4. If the P ay
icon in the
open to j

5. Play back
and the o
you don't s
MIDI ch n
notes in th

THE MI

The MIDI M
with 16 c ff
combinatio
the same MID
represente b
right column
in the right o

As long as yo
Mixer kee s
Keyboard to s
then set the M
the MIDI Mix
Keyboard o
you connec t
MIDI channel
instead of s t
module for ch
will mix the v

4. If the Player Piano window isn't open, double-click on the Player Piano icon in the Patch Panel to open it. (The Set Up Player Piano window will open too, just click on its close gadget to close it.)

5. Play back the score. You'll see the notes in the first track displayed in grey and the notes in the second track displayed in red on the Player Piano. (If you don't see them, check to make sure you have the Player Piano set to MIDI channels 1 and 2, and that it's using the right octave span to show the notes in the track.)

THE MIDI MIXER MODULE

The MIDI Mixer is a simple and very useful module. Think of it as a cable with 16 different wires, one for each MIDI channel. Any signal or combination of signals you plug into a MIDI channel at one end comes out of the same MIDI channel at the other end. The two ends of the MIDI Mixer are represented by two icons in the Patch Panel; the MIDI Mixer input icon in the right column accepts signals from other modules, the MIDI Mixer output icon in the right column sends out the signals it receives through the input icon.

As long as you put signals into the MIDI Mixer on different MIDI channels, the Mixer keeps the signals separate. For example, if you set the Console Keyboard to send data on MIDI channel 1, connect it to the MIDI Mixer input, then set the MIDI In module to send data on MIDI channel 2 and connect it to the MIDI Mixer, the MIDI Mixer output sends out signals from the Console Keyboard on channel 1 and signals from the MIDI In module on channel 2. If you connect two signals sources to the MIDI Mixer that both use the same MIDI channel, the Mixer will combine the two signals. For example, if, instead of setting the Console Keyboard for MIDI channel 1 and the MIDI In module for channel 2 you set them both for channel 1, the MIDI Mixer output will mix the two signals together and send them out on MIDI channel 1.

You don't really need the MIDI Mixer for connecting and mixing signals in the Patch Panel. You can connect two output modules to the same input module to automatically mix signals. For example, you can connect both the Console Keyboard and the MIDI In icons directly to the Sampled Sound Player. If they're both on the same MIDI channel, the Player will automatically mix the signals. The real utility of the MIDI Mixer is to make connections with signals from the Tape Deck.

You can choose the MIDI Mixer as either the input or output module for any track on the Tape Deck. If you set the MIDI Mixer as the input module for a track, any modules you connect to the MIDI Mixer input icon in the Patch Panel will go through the Mixer and come directly into the Tape Deck. If you set the MIDI Mixer as the output module, the output signal of the track will go to any module that you connect to the MIDI Mixer output icon in the Patch Panel. The following example uses the MIDI Mixer with the Tape Deck to combine two tracks into a third:

1. Create three new tracks in the Track List in the Tape Deck window.
2. Record two different sequences, one on each of two of the new tracks. Leave the third new track empty.
3. Set the output mode of each of the newly recorded tracks to either Thru or "blank".
4. Set the output module for each of the recorded tracks to the MIDI Mixer, and set the MIDI output channel to 1.
5. Set the operating mode of the recorded tracks to Play.
6. Set the operating mode of the blank track to Record.
7. Set the input module for the blank track to the MIDI Mixer, and set its MIDI input channel to 1.

8. Click the R button on if it isn't on already.

9. Click on the Play From Beginning button to start the sequences playing.

10. Click on the Stop button when the sequences are done playing.

When you try this example, the Tape Deck plays the two recorded tracks into the MIDI Mixer on MIDI channel 1, where they are mixed together and fed directly into the third track, which records them. Since the number of tracks in the Tape Deck is only limited by memory, it isn't always necessary to mix down two tracks like this, but it can be handy if you want to keep the number of tracks down to a manageable number.

THE MIDI IN MODULE

The examples so far in this manual have only used the Amiga's own keyboard and sound synthesis capabilities. SoundScape will also accept signals from and send signals to any external MIDI device attached to the Amiga. You can use SoundScape to record performances on external MIDI keyboards, to play back Tape Deck scores using external MIDI synthesizers and drum boxes, and to control networks of external MIDI devices by using the MIDI In and MIDI Out modules.

The MIDI In module passes on any MIDI signals that come in from an external device through the MIDI In port of the Amiga's MIDI adaptor into SoundScape's Patch Panel. You can connect the MIDI In module to any other module to receive those signals. For example, if you want to record a performance on an external keyboard in the Tape Deck, you select the MIDI In module as the input module for a track. As you play on the keyboard, the track receives the keyboard's MIDI signals. (That is, it does if it's set to receive on the same MIDI channel that the keyboard is sending on.)

External devices usually send out MIDI signals on just one MIDI channel, usually the same channel they use to receive MIDI signals. Any module you connect to the MIDI In module in the Patch Panel will receive MIDI signals from the external device on the same channel. If you want to change that channel, you can change it directly at the device. Some MIDI devices have a fixed output channel (usually channel 1). You can get around the fixed channel by offsetting channels that come in through MIDI In: Double click on the MIDI In icon in the Patch Panel to open the Set Up Midi In window.

In the Set Up Midi In window is a single slider: the Channel Offset slider. You can set it to offset incoming MIDI channels by the amount on the right end of the slider. The slider is usually set to 0, a setting that lets all the channels come into Patch Panel on the same channel that the external device sends them on. If you set the slider up or down, you change all 16 incoming channels. For example, if you set an offset of +2, the incoming MIDI channels are changed like this:

Channel 1 become channel 3
Channel 2 become channel 4
Channel 3 become channel 5
Channel 4 become channel 6
Channel 5 become channel 7
Channel 6 become channel 8
Channel 7 become channel 9
Channel 8 become channel 10
Channel 9 becomes channel 11
Channel 10 becomes channel 12
Channel 11 becomes channel 13
Channel 12 becomes channel 14
Channel 13 becomes channel 15
Channel 14 becomes channel 16
Channel 15 becomes channel 1
Channel 16 becomes channel 2

You can use the channel offset to take an external device sending on MIDI channel 1, for example, and change it to come into SoundScape on MIDI channel 9.

THE MIDI OUT MODULE

The MIDI Out module is arguably the simplest module in SoundScape: Any MIDI signals sent into it in the Patch Panel or from the Tape Deck pass directly out through the MIDI Out port of the Amiga's MIDI adaptor to attached external MIDI devices. If you want to send MIDI messages to any particular external device, you can set the device's MIDI channel number to an exclusive channel, and set the sending MIDI channel of the messages you send to the same channel number. For example, say you want to send a Tape Deck track's playback to an attached external synthesizer. You can set the synthesizer to receive messages on channel 12, for instance, and then set the track's output module to the MIDI Out module using output channel 12. When you play back the score, that track will send messages directly to the synthesizer, which will receive them and play them.

THE CLOCK MODULE

Although the clock's primary purpose is to provide timing information, it can also receive some. If the clock is not currently running, it can be started by sending it a Start from Beginning or Play event (from the Console Keyboard, for example.) It can be stopped with a Stop event. And its tempo can be sped up or slowed down. Here's a good way to demonstrate this last one:

1. Load a score into the Tape Deck.

2. Create a new track and put it at the top. Set the input and output modules to the Clock. Open the Status In window and turn on Control Change. Do the same with the Status Out window.

3. Set this track to Record mode and turn on the R button in the Tape Deck.

4. Hit Start from Beginning on the Clock.

5. As the song plays, speed it up and slow it down by moving the tempo slider on the clock.

6. Stop the song and turn off Record.

7. Open the Sequence edit window and you should see a list of control change events. These correspond to the tempo changes.

8. Play it. Waddyaknow, it speeds up and slows down.

SAVING SEQUENCES, SCORES, AND SONGS, AND ENVIRONMENTS

You can use the Save command in the Edit Sequence window to save a sequence to disk, the Save command in the Edit Song window to save a song to disk, the Save command in the Tape Deck window to save a score to disk, the Save command in the Patch Panel to save an environment to disk, the Save command in the Sampler window to save an instrument to disk, and the Save All command in the Sounds window to save an instrument bank to disk. You can then use the Load (or Load All) command in the same windows to load the same type of file. What's the difference between all the different saved files and how do they load back again?

Well, saving an instrument in the Sampler window saves the samples stored in the Amiga's memory and the settings of all the controls in the Sampler window. Loading an instrument puts the samples back in memory and sets the Sampler controls back to their saved settings. Saving an instrument bank saves the names of all the different instruments in the 16 different channels. Loading an instrument bank puts the names of all the different instruments

back into the channels, and then loads the instruments themselves (including samples and control settings.)

Saving sequences, songs, score, and environments are like taking closeup and group photographs of people. Saving a sequence is a closeup of one element of SoundScape music: a sequence of MIDI events stored in the Event List of a single Tape Deck track. Saving a song steps back to include more SoundScape elements in the picture: SoundScape saves all the different sequences and their transpositions that constitute a song in a single Tape Deck track. Saving a score takes an even more inclusive picture of SoundScape elements: SoundScape saves all the tracks in the Tape Deck Track List, including all the track settings and all the songs in each track, which include all the individual sequences within each song.

Saving an environment is like taking a full group picture of SoundScape: SoundScape saves the score in the Tape deck (which of course saves all the tracks with all their songs, settings, and sequences), the instrument bank in the Sounds window, and all the control settings of each of the modules (like the MIDI channel settings and function key assignments of the Console Keyboard module).

When you decide to save music in SoundScape, you can choose any level of detail. Saving an environment is best for a finished piece of music set to play on specific instruments, because you have all the connections made for instant playback. Saving scores is handy for a piece of music that you might want to play back using different instruments or external MIDI devices. You can think of songs, sequences, instrument banks, and instruments as building blocks that you can save to use in other pieces of SoundScape music you compose.

ADDITIONAL SOUNDSCAPE MODULES

SoundScape is written to be very flexible and expandable. The modules included on the SoundScape disk are not the only modules that will work with SoundScape. As time goes on, you'll find additional modules for sale that add to the power you already have in SoundScape. When you load these modules from their disk, SoundScape will automatically add icons for them in the Patch Panel so you can connect them to any other SoundScape modules. As you add modules, you can resize the Patch Panel window to accomodate all the icons.

As you make music with these new modules and want to save your work, you can always use the Save command in the Patch Panel to save an environment. SoundScape will then save all the settings in the new modules as well as the settings in the existing SoundScape modules. If you should ever try to load an environment that has saved settings for modules that you don't have, the environment will load anyway, ignoring the saved data for modules that you don't have running.

SoundScape's modular construction makes it flexible and expandable enough to give you an infinite variety of configurations and creative possibilities. Your only real limitation is your imagination.

IFF SMUS and SoundScape

Infact, the very first SoundScape module is a program, SMUSConvert, that takes SoundScape scores and converts them into the IFF SMUS format. SMUSConvert will convert from IFF to SoundScape as well. As a SoundScape module, SMUSConvert requires SoundScape in order to run. If run first, it will invoke SoundScape, so don't be surprised when the Patch Panel pops up.

SoundScape

PRO MIDI STUDIO

ADDENDUM

mimetics™

corporation

... the Professional Software Source!!

Table of Contents

Introduction

We've put some new features into SoundScape. These notes cover these features and conclude with a discussion on how SoundScape is implemented as a "library" on the Amiga.

Multiple Edit Windows

The SoundScape Patch Panel now supports opening multiple edit windows. Previously, if an edit window was opened from the Patch Panel, all further Patch Panel activities such as opening another edit window or connecting two modules, were not allowed until that edit window was closed.

Now, you can open an edit window (by clicking twice on an icon) and continue to work with the Patch Panel.

Directory Support in the Load Requester

When you execute a LOAD command (be it environment, samples, tracks, etc.), it finds all the files of the proper type in the selected directory. It now also includes the names of subdirectories (drawers) that exist within this directory. These subdirectories are labeled with "`(dir)`" after their name. To access a subdirectory, click on its name twice. SoundScape will search this subdirectory and display the files found. In addition, the subdirectory name is now appended to the directory name above the file name list. To leave this subdirectory, simply click on the directory name and edit it out.

Default Disk for Loads and Saves

When you first execute a load command, you will notice that, instead of looking for files in the volume "SoundScape:", it now looks in the volume "Default:".

With AmigaDos, you can assign any arbitrary volume name to any directory. So, if AmigaDos has been told beforehand that "Default:" is, infact, "SoundScape", it will automatically look there.

The AmigaDos CLI command to make this assignment is called "Assign". So, if you typed

```
Assign Default:  SoundScape:
```

from the CLI, from then on the name "Default:" could be used as a name for the volume "SoundScape:".

If you ran the installation program for your SoundScape (be it an update or the new product), it set up the modules file (see the section "New Modules File") to do the above assignment. So, when you do your first Load and it displays "Default:" as the directory instead of "SoundScape:", it still reads the files from your SoundScape disk.

What does this give you? You can now change your default directory to be something other than "SoundScape:". This comes in real handy if you have a hard disk.

Beats and Measures

With the SoundScape Sequence and Song editors you can now edit time values by beat and measure, instead of just by clock ticks.

Time Signature

In the Edit Sequence window, there is a new string gadget:

```
T Sig 2/4      B
```

By editing this, you decide what time signature the sequence should be displayed in. The number on the left side of the slash (in this case, 2) is the number of beats per measure. This can be any number from 1 to 99. The number on the right hand side of the slash says what type of note is used to define a beat. For example, 4 indicates a quarter note, 8 an eighth note.

A quarter note is 24 MIDI clock ticks.

You might prefer defining the beat not by the type of note but by the duration of the note in MIDI clocks. For example, although a quarter note can be used to represent 24 clocks, there is no note that represents 25 clocks to the beat. Should you be so perverse as to desire your song broken into measures defined by nonstandard notes, click on the 'B'. It turns to 'C', indicating you can now define the beat in clocks. So, 4/4 turns into 4/24. We aim to please.

If you would rather stick with clock values, just set either number to 0.

Interpreting the Time

With beats and measures, time values are displayed like this:

`measure:beat:clocks`

Example:

`10: 1: 12`

This says, "We're ten measures into the sequence, in the first beat of that measure, and we've advanced by 12 clocks in the beat."

Measures and beats always start with 1 (the first beat or measure) and count up. Clocks start from 0. So, `1:1:0` actually is the very start, being the first measure, first beat, with no clocks so far. This might be confusing at first, so play around with it a bit.

When you edit a time value, if you put in numbers that don't make sense, it should reject the edit and display the original value.

There are four places where you actually edit time values. In the Edit Sequence window, there are two:

`Time 10: 1:12`

`Next 10:20:0`

`Time` specifies the moment in time the current MIDI event occurs. Should you edit it, you will reposition the note without altering any of the others in the sequence.

`Next` defines when the note that follows should occur. Changing this will shift all notes that follow by the same amount. Unlike older SoundScapes (and the manual), this is an absolute value. In older SoundScapes, this was a relative value. You specified how long after this event the next event occured. Now this number is simply the actual time of that event.

Should you attempt to change the next time value to a value prior to the current note, it will ignore your attempt.

In the Song window, there is one place you can edit a time value:

`Start 19: 4:0`

This used to be called 'Wait', because it defined how long to wait after the end of the last sequence before starting this sequence. Now it simply defines in absolute terms when the first event of the sequence occurs.

Finally, from the Tape Deck itself, we can edit in an initial delay:

```
Delay   20:  1:0
```

The Move Command

The Edit Song window has an additional feature, the 'Move' command. You can use this to rearrange songs. Move a sequence around within a song by first clicking on it, then clicking 'Move', then clicking on the position in the song list you'd like the sequence to be.

Edit Positioning

If you play, rewind, or fast forward to a particular point in your song, then enter either the Sequence or Song editor, the cursor will automatically go to that current position in the song.

Auto Insert Disable

You can enable and disable the Punch In and Punch Out registers. To enable the Punch In register, click on the 'PI' button to the left of it. The button will turn red, indicating it is enabled. Then, when you set the Tape Deck running, it will enter record mode when the location indicated in the Punch In register is reached. Disable Punch In by clicking again on the 'PI' button.

You can enable and disable the Punch Out register by clicking on the 'PO' button.

MIDI Insert Editing

You can now enter MIDI events into the sequence editor by playing them from a keyboard. This can be the console keyboard, an external MIDI keyboard, or any other MIDI device. Here's how to use it:

- Select the track you wish to edit in the Track List.
- Set the track in RECORD mode. This enables insert editing.
- Click on "ST IN" to edit which types of events you wish to insert. Generally, you would only be interested in Note events, so disable all others, such as Pitch Wheel.
- Make sure the device you wish to enter notes with is set up as the module in for this track, with the proper channel in set.
- Click on "Sequence" to enter the Sequence Editor.
- Select where you wish to edit. Do this by selecting a note in the note list. The inserted note will be put there (without deleting the selected one.)
- Play a note. It will be inserted and displayed.

If you append a note at the end of the list, it will be placed after the previous last note by the amount defined in the "Next" field of the last note. In this way, you can quickly enter notes in step time. Try it. It's great for designing arpeggios for Echo mode, among other things.

If the note is entered somewhere in the middle of the list, it doesn't adjust the timing. Instead, it has this note occur at the same time as the selected one, creating a chord. You can now edit it if you wish to put it somewhere else.

Undo

You are now given the ability to enable and disable the Undo command. In order for Undo to work, copies have to be made of all sequences prior to recording. This eats up a lot of memory, especially if you're recording lots of pitchbend and aftertouch. So if you don't plan to use Undo, it's often wise to disable it.

To alternately enable and disable Undo, click on the new icon placed just to the right of the Undo button.

By default, Undo is disabled.

Panic Button

It is possible to gridlock SoundScape.

For example, you could have a whole bunch of modules hooked up with a feedback loop somewhere in there. A note would enter this feedback loop and cycle infinitely, resulting in the system acting as if it were hung (but it's really just doing its job.)

When this problem occurs, all the windows can be moved and icons clicked, but nothing new happens.

There's a way out. Hidden behind the Patch Panel is a small window with "Panic Button" displayed in red. You can get at this window by simply moving the Patch Panel out of the way or moving it to the back.

Click on the red panic button. It will clear out all currently transmitting events and tell all modules to stop.

There are other ways of hanging the system. You could describe a sequence that loops with no time increment between events. When playing, the Tape Deck would cycle infinitely, outputing the same events over and over again. It's actually hard to trick the Tape Deck into doing this because the Sequence Editor looks out for such loops. But, if you are determined, you can always find a way. The Panic Button will pull the Tape Deck out of such a loop.

Or, you could simply have too much going on. Try to drive SoundScape mad by turning the clock up to full blast and then running it through a rate scaler module that turns it up even higher and have everything running and processing events at once. If enough is going on, the computer can no longer get everything done on time and the system gets farther and farther behind. The Panic Button will halt this insanity.

However, you probably will never need the Panic Button. It's just nice to know it's there.

SMUSConvert

SMUSConvert is a program that converts SoundScape music files into the IFF SMUS file standard and back.

SMUSConvert runs with SoundScape. You can invoke it while SoundScape is running or beforehand. If you invoke SMUSConvert first, it will start SoundScape (see the section on SoundScape as a Library.)

Converting from SoundScape to SMUS

Under "SoundScape→SMUS" is two text gadgets. With "Quarter Note" you define how many MIDI clocks define a quarter note. It defaults to 24, which is the MIDI standard. You will probably rarely change this.

"Quantization" defines what the resolution of the notes should be. With the SoundScape file format, each note can have any random duration. With SMUS, each note is defined within traditional notational constraints. So, a note with a duration of 18 clocks would be a dotted eighth note. Fine. But a note with a duration of 21 clocks would be a dotted eighth tied to a thirty second note. Things get messy and the rests between notes also need to be defined in this way.

So, there's a real need to quantize the time a note occurs and its duration to a reasonable value. The value is defined in clocks. The default is 12 clocks, or an eighth note. With this quantization, the above example would give you an eighth note for the 18 clock duration and a quarter note for the 21 clock duration. If higher resolution is required, set the quantization number appropriately. A value of 6 would result in resolution down to sixteenth notes. To define a resolution, remember there are 96 clocks to a whole note. Simply divide 96 by the note type to get the number of clocks. 96 / 4 is 24 clocks per quarter note. 96 / 16 is 6 clocks per sixteenth note.

If you define a number that doesn't divide into 96, like 13, it will not define any legal note type. The results will probably be strange and useless.

Play around with different resolutions and see what they generate.

Having set the quantization and quarter note values, click on "SoundScape → SMUS". A standard load requester will open, asking you which file to load. This will be loaded and converted to IFF. Then another requester will ask you what the file should be saved as. '.SMUS' will be appended to the file name as it is saved.

If there are tracks in your score that you would rather not have converted into SMUS, such as a drum track, you can disable conversion of it by setting it Mute when saving to disk.

All looped sequences and song tracks are played out until the score's end, which is defined by the longest non-looped sequence or song.

SMUS is not a performance format. Pitch wheel, after touch, etc., are removed.

Converting from SMUS to SoundScape

Under the "SMUS→SoundScape" icon, there is another text gadget that asks you to define how many clocks there should be per quarter note. It defaults to 24, which is the MIDI standard.

If the file does not have ".SMUS" appended to the end, you will need to type in the whole name of the file. The box under "SMUS File Name" lets you do that.

Click on "SMUS→SoundScape". If you didn't specify a file name, a standard load requester will prompt you for one. The file will be loaded and converted to the SoundScape format. You will then be asked to give a new file name and it will be saved.

New Modules File

Along with the modules that come initially with SoundScape, additional ones can be bought and installed so that they become part of the SoundScape system, appearing in the Patch Panel when SoundScape starts running.

In order for these modules to come up as part of SoundScape, SoundScape has to find them and know to install them. This is accomplished by having a file with a list of module commands in it. On start up, SoundScape opens this file and reads the module instructions from it.

Up until now, this file has been `SoundScape:ss/modules`. But this means reading and occasionally updating a file on your master copy of SoundScape, something it would be nice to avoid. So, an alternative modules file can now be used. It is identical in format to `SoundScape:ss/modules` but can exist on your WorkBench disk. Name it `SS-Modules` and place it in the `S` directory of your WorkBench disk.

When looking for modules to load, SoundScape will first search for `S:SS-Modules`. If it can't find it, then it will try `SoundScape:ss/modules`.

The format for a modules file should be:

```
run modulename
run anothermodulename
run yetanothermodulename
```

In other words, it is simply a list of CLI commands to execute.

To create and edit this file, use a text editor such as EMACS or Edit.

For those who aren't too comfortable with the idea of editing a CLI command file, we'll be providing a simple program that lets you select which modules you want to autoload. It will change the modules file for you.

This program will be included with the modules.

SoundScape as a Library

SoundScape is not implemented as a single program that you run (although it may appear that way.) Instead, it is implemented as a "Library" on the Amiga. This gives SoundScape much of its power and flexibility.

But it can also cause some confusion, because things aren't quite what you'd expect. So, here's a brief explanation of what's going on:

You don't run SoundScape directly.

Instead, you run a program that invokes SoundScape. This program asks the Amiga's operating system to start SoundScape running. The Amiga's operating system looks for `soundscape.library` in the `libs` directory of your WorkBench disk. If it is there, it loads it in and starts it running. The Patch Panel opens and the modules are displayed.

Then, the program that asked for SoundScape can now work with SoundScape. For example, it can install its icon in the Patch Panel and start sending and receiving notes. In addition, multiple programs can invoke SoundScape at the same time, all working with SoundScape in one way or another.

The very simplest program that uses SoundScape is, infact, also called "SoundScape" (we'll refer to it in quotes), and it is what you are probably familiar with. But if you were to inspect it, you would notice that it is a very small program. All it does is put up a window (that one in the top left corner) and ask the Amiga to start the real SoundScape running. Then it waits for you to close its window. When you do, it simply tells SoundScape to shut down.

But the real SoundScape program that you see doing all the magic with the Patch Panel, Sampler, Tape Deck, etc, is actually `soundscape.library`.

Another example of a program that invokes SoundScape is SMUSConvert. When SMUSConvert starts running, it asks for SoundScape to be run. Once again, the Amiga's operating system loads in `soundscape.library` and sets it running. Then, SMUSConvert uses it to access music files and convert them.

If you run both "SoundScape" and SMUSConvert, you will notice that SoundScape gets invoked only once, and that you need to shut both down (in any order) before SoundScape will close.

Fortunately, all of this library stuff is transparent to you, the user.

But, because of this transparency, if SoundScape doesn't run, you may scratch your head and wonder why. If that happens, here's a checklist of criteria that must be satisfied in order for SoundScape to run:

- Your WorkBench disk has a copy of `soundscape.library` in its `libs` directory. This is done automatically by the program SSinstall, which you run when you first get SoundScape. To run SSinstall, simply reboot with the SoundScape disk as your WorkBench disk. It will tell you what to do.

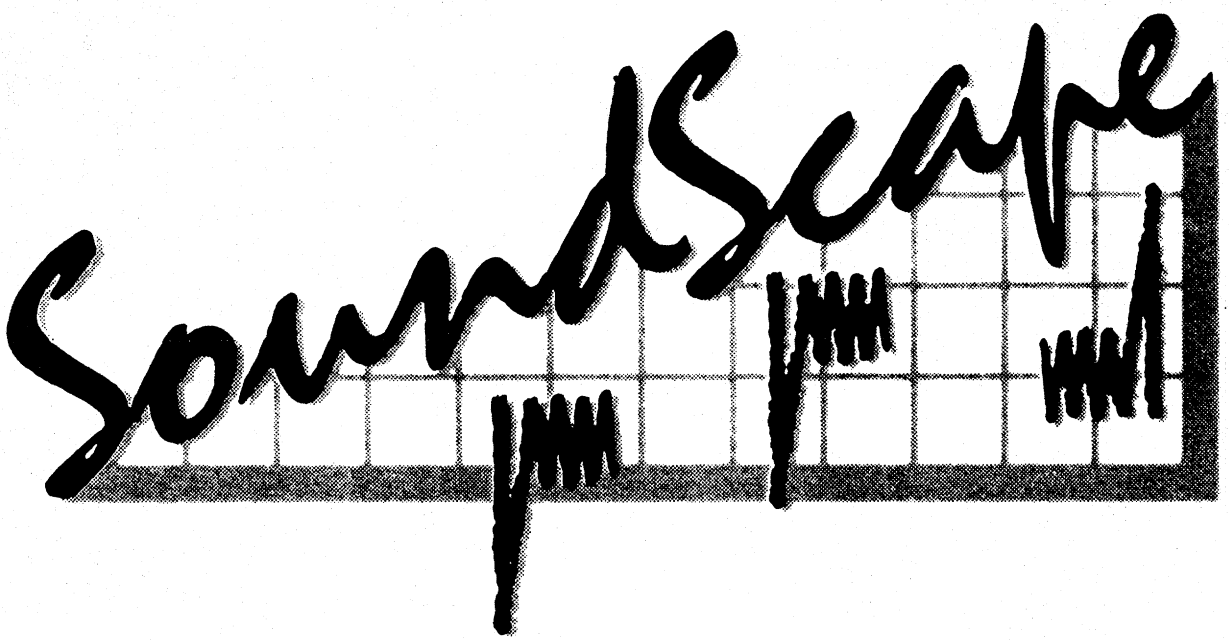
- You must also have `mathtrans.library` installed in the `libs` directory. This is also handled by SSinstall.

- The WorkBench disk that has `soundscape.library` is also the WorkBench disk you booted with. If you simply swap disks and then invoke SoundScape, the Amiga will ask for the WorkBench disk you booted with, whether or not it has `soundscape.library` in it. If it doesn't, it won't run SoundScape.

- You have a program, such as "SoundScape" or SMUSConvert that can invoke SoundScape.

- There's enough memory. SoundScape will not run on 256k machines. If you are running on a 512k machine with several other programs going, there might not be enough room to load `soundscape.library`.

One important ramification of all this: When you update your SoundScape to a new version, you should be updating `soundscape.library`, not the program "SoundScape". And, make sure you do the update on all WorkBench disks that you are using for SoundScape. Otherwise, depending on which you booted up with, you may or may not be running with the latest version of the program, even though you installed it on the SoundScape disk.



Sound Sampler

ADDENDUM

mimetics™

corporation

...the Professional Software Source!!

Table of Contents

How s

Simply

Overvie
to use h
section d
the follow

Captu i
used whe
rather ir
like a r ic
yellow, cl
up righ ,
section o
icon used
they ar

Editing
once you
such to i
points; ar

Files - I
formats

Referen
the Sar p

How is This Document Organized?

Simply:

Overview - This, the first section, is intended to shed some light on how to use the Sampler Software. The nuts and bolts that make it tick. This section doesn't have pictures, just ideas. The play-by-play stuff comes in the following sections.

Capturing A Sample - This section dwells on the options that can be used when recording a sample of sound. Actually, recording is really a rather simple process. All you need to do is click on the icon that looks like a microphone, click the left mouse button, and, when the screen turns yellow, click the button again. If you do that, and if the mike is hooked up right, you'll record something. So, since recording is that simple, this section focuses on the things that you can do to enhance the recording. Each icon used to record a sample is illustrated. Each is discussed. Moreover, they are presented in the order they would be used.

Editing A Sample - This section dwells on editing a sample of sound once you capture it. It is broken down into several subsections covering such topics as: creating octaves; editing the waveform; establishing loop points; and creating amplitude envelopes.

Files - This section is brief. It covers saving and retrieving files and file formats.

Reference Section - This addresses some of the more technical aspects of the Sampler software. The various topics appear in alphabetic order.

Overview

The sampler is a hardware device that plugs into the second mouse port of the Amiga computer. It can be used with a microphone or a line level source to digitize the sounds of a musical instrument, voice, or any other sound in a way that the can be played back later in any arrangement of harmony or cacophony.

The hardware device actually digitizes the sound, and the Sampler software captures it.

In its digitized form, the sound can be edited, manipulated, and stored in a file for use by SoundScape and other Amiga software.

There are a few options you can set to manipulate the recording process. And, of course, there are a few different ways of manipulating the sound after you capture it.

The options you can set for the recording process include:

- You can select a high or low sampling rate - how many thousands of times a second the Sampler reads the sound.
- You can manipulate the volume at which the sample is recorded. You can record in a compressed mode, which serves as an automatic gain control. Or you can record in noncompressed mode, and set the volume yourself.
- You can set a volume threshold, so that the Sampler does not begin recording until the source sound reaches a certain volume.
- You can set the length of the recording. The actual length of the recording is influenced by two factors. One is the amount of memory you allocate to the sample buffer. The other is the sampling rate you choose. A high sampling rate and a 65K buffer (the maximum) yields roughly a five second sample. A low sampling rate with the same size buffer yields a nine second recording.
- You can select which one of the 10 octaves you want to record into. The default is octave 5. After the sound is captured, it can be translated into other octaves.

After you select the recording options you want to use, the recording process does not begin until you click the microphone icon. Then the screen will

turn red. The screen remains red for about two seconds, then turns yellow, indicating it is ready to record. Now you have to click the left button on the mouse and recording will begin when one of two things happen: when the volume reaches the volume threshold, or when you click the left mouse button a second time. At this point, the screen turns green, indicating it is recording.

When the recording is complete, the Sample Capture & Edit screen displays the waveform of the sound. At this point there are a number of different things you can do to edit the sound.

- The first thing you may want to do is listen to the recording. If you recorded the sample into octave five, you can play it back by pressing Y on the Amiga keyboard. If you have recorded it into some other octave, you must press the key that corresponds to that octave. The sound can be played back at any time during the editing process. Remember, the sound does not exist in any octave other than the octave it was recorded into. It does not exist in any other octaves until it is converted using a octave translation procedure. More about that later.

- You can edit the sample's waveform by redrawing the waves. Using this graph, you can also set loop points so that the sound holds and repeats the parts you like.

- You can also convert the sound to additional octaves to fill the full range of the keyboard.

- Finally, last but not least, you can save your work in a file. This disk file can be in one of two formats. The Mimetics format, for use by the SoundScape software, or the IFF format, an interchange format that enables other software and games running on the Amiga computer to be able to use your samples.

Once a sound is recorded, it can be played back immediately. Simply press the keys associated with the octave you are working in. It can be played back at any time during editing.

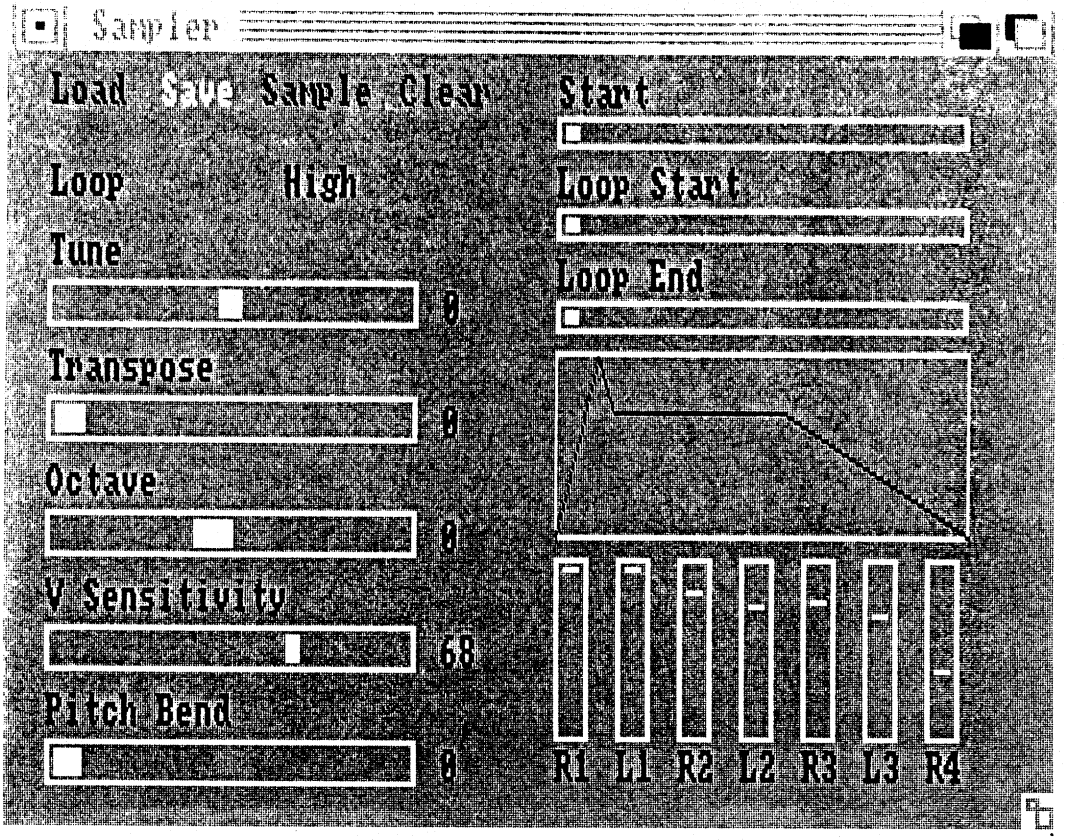


Fig 1. The Sampler window, the first window displayed.

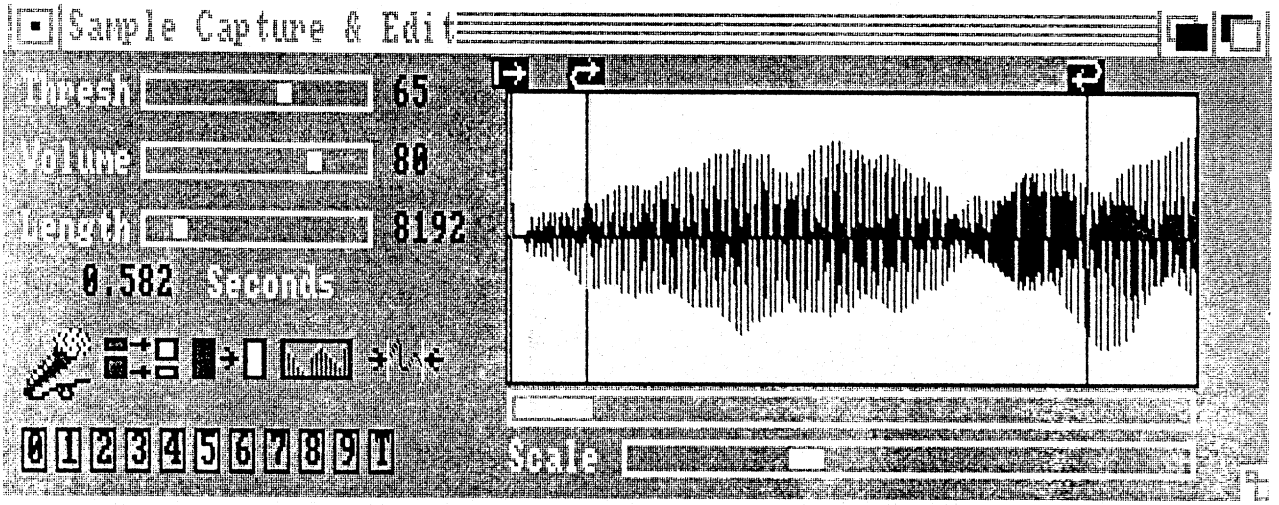
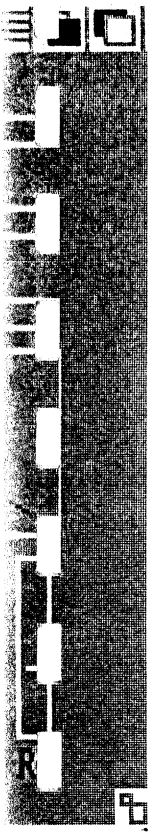


Fig 2. The Sample Capture & Edit window.



la ed.

Capturing A Sample

This section covers the options available for the recording process. Each option is followed by discussion, and appears in the order it would be used.

The following icon appears on the window named Sampler. The remaining icons appear on the Sample Capture & Edit window. To enter the Sampler window, click on the Sampler icon.

High/Low Sample Rate

This icon appears on the Sampler window, the first window displayed.

Low High

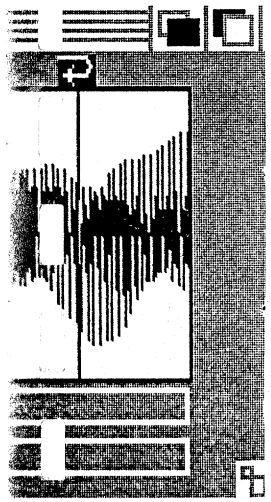
It sets the sample rate. Click on the icon and see how it toggles between low and high.

If you select High, the sound is recorded at 14,080 times a second, roughly 14 Khz.

If you select Low, the sound is recorded at 7,040 times a second, roughly 7 Khz.

The high sample rate yields the highest quality recording and a five second sample per 65K buffer. The low sample rate yields a nine second sample per 65K buffer.

The following icons appear on the Sample Capture & Edit window.



Octave

There are ten octaves. Select which octave you want to work with. The number of the octave you select appears in white. This is the active octave.

If you select octave five, you will be able to play the sample by pressing the keys Y through] or by hitting the caps-lock key and by playing Q through 8. Each octave is from A to G#. The console keyboard supports the upper half of octave four, all of octave five, and the lower half of octave six. These and all the other octaves are supported by MIDI.

If you are only using the console keyboard, you might consider always recording into octave 5, then copying your sample into the other octaves when you are done.

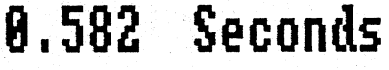
Length

This sets the length of the sample buffer. This buffer is used to capture the incoming sounds, and when it fills up, the recording is complete. The shorter the buffer, the shorter the sample. The buffer can be up to 65K in size.

Set the amount by sliding the box. If it refuses to allocate a large buffer, it means you are running low on memory. Shut down any small programs or windows you have running in the background. Or, use a smaller buffer.

Seconds

This indicates the duration in seconds the buffer can hold. This figure appears automatically.



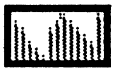
Two things affect the duration of a sample:

- High and low sample rates.
- The amount of memory allocated for the sample.

Compressed/Non-compressed

You can smooth out volume differentials from the source by recording in compressed mode. This cuts down the loud sounds, and brings up the quiet ones. It serves as an automatic gain control.

The symbol for Compressed is a straight line. The symbol for noncompressed is a mountain. Click on the symbol to see how it toggles between the two images.

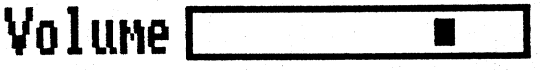


Recording in noncompressed mode lets you set the recording level yourself using the volume slider.

When you record in the compressed mode, the volume slider only sets the level for the start of recording.

Volume

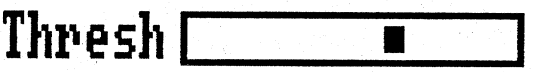
This sets the recording level when you are recording in noncompressed mode.



In compressed mode, this slider sets the initial recording level before recording starts.

Threshold

After you click on the microphone icon, the system waits for the source sound to reach a certain volume. Then it starts recording. This sets that threshold.



Recording

Ready, set to record ... click the microphone icon ...

After you click the mike, the screen turns red. It stays red for about two seconds, then turns yellow. At this point, click the left button of the mouse and recording will begin when one of two things happens: when the volume reaches the volume threshold, or when you click the left button of the mouse a second time.

When recording commences the screen turns green. When it finishes, the Sample Capture & Edit screen reappears, displaying the waveform of the sound just recorded.

If it does not enter the record mode, either the microphone is switched off, you have nothing plugged into the Sampler hardware, the Sampler is not properly installed, or the volume threshold is set too high. If you are waiting for a threshold, and you realize that the volume won't trigger the recording, no problem. Just click the mouse, let it record, reset the threshold, and do it again.

Editing a Sample

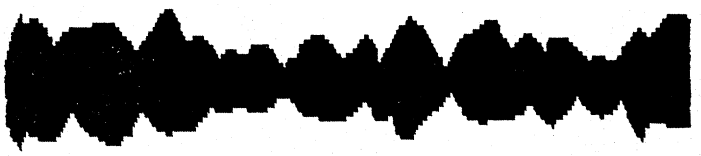
Once you have captured a sample, there are a number things you can do to edit it. You can:

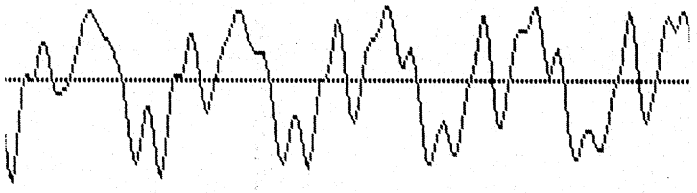
- View the waveform in its entirety or in any resolution down to individual cycles.
- Edit individual waves.
- Establish loop points so that the sound holds and repeats the parts you like.
- Cut the sample down in size so only the part that is played is retained.
- Translate the sample to other octaves.

Viewing The Waveform

The waveform is an editable graphic representation of the sound sample. In this format, the sound can be viewed all at once or in pieces.

Viewed all at once, the sound can look like a silhouette of the Rocky Mountains.



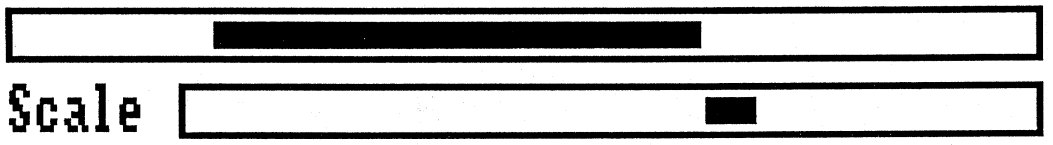


In medium resolution, it can look like the hills back home. In fine resolution, it looks like a fishing line amid cast.

Scale & Position

The scale slider indicates how much of the waveform to display. Setting the Scale all the way to the left gives you the finest resolution. Setting it all the way to the right gives you the whole thing.

The upper slider moves the display. Its size changes depending on the setting of the scale.



Editing the Waveform

With the scale all the way to the left, the individual cycles can be edited.

Click on the left mouse button and draw the cursor across the lines. See how they change.

Start Playback Point and Loop Points

There are three icons above the waveform that look like arrows. These icons slide back and forth and mark the start playback point and the loop points.

To change a point, grab the icon and pull it into position. Once you let go, it is set.

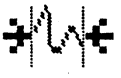
The icon with the straight arrow indicates the start playback point. This is where the sample begins playing.

The two icons with the right and left arrows set the loop points. Looping does just that. It causes the sound to loop and hold until it fades out.

Unless the waveform is in fine resolution, the points are moved forward to the first downward zero crossing. This insures glitch free transitions. In fine resolution, they stay where you put them.

Truncate

After you finish editing and are satisfied with the sample, click the truncate icon. This economizes on the amount of memory consumed by the sample. The sound before the start playback point and after the loop end point is deleted, and that freed memory is returned to the system.



Translating Octaves

You can create up to ten octaves from the same sample of sound by converting the sample up and down to the other octaves.

There are two ways to convert a sound from one octave to another. One actually changes the sound into that fitting the octave you select. The other way simply copies it without making any changes.

Indicating What Octaves to Work with

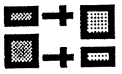
Look at octave numbers. One is red. One is white. Red converts to white. Red is always the last octave you worked on, white is the current one.

If you were to click the conversion icons right now, the sound in the octave with the red number would be converted into the octave with the white number. Click the octave numbers, and see how the colors change. Now that you know how to indicate the direction of a translation, you are ready to select a type of conversion. Read on.

(By the way, the T indicates a temporary buffer, which can be used to copy an octave to or from another channel.)

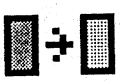
Octave Translation

This converts the sound to the octave you choose. For example, if you translate from octave five to octave four, you would get the same sound, just one octave lower. Or, if you were to translate from octave three to octave eight, you get a sound five octaves higher, while leaving octaves four through seven alone.



Simple Conversion

This simply copies the sound without changing it in any way.



Files

Reading and Deleting Files

From the Sampler window click the Load command, and a new window will appear with a list of the sample files in the SoundScape directory. At this point you can select a file, delete a file, quit, or select another directory.

There is a slider to the right of the file names. It will let you view more file names, should there be more in the directory than can be displayed at one time.

Click twice on the name of the file you want.

If you want to look in another directory, edit the name of the directory. It defaults to "SoundScape:".

Saving Files

When you click the Save command, you can save your work in a file, or simply quit and return to the Sampler window. To save the file, you must type its name and then hit the return key.

IFF versus Mimetics File Formats

The Mimetics file format should be used at all times. Use the IFF format only if you are creating sounds to be used by other programs that support the IFF file format.

Not all the data about a sound is saved in the IFF file format. Performance information is thrown away. In addition, IFF assumes that the same sample is used in all ten octaves, or however many you have filled with samples. This is fine if you recorded one sound, set its loop points, then translated it up and down to the other octaves. However, if you have various length samples and loop points, you might not be happy with the results.

The IFF format assumes that the sample has one size, which is halved as you go up octaves and multiplied by two as you go down. The sample can have one loop start point and one loop end point. When a file is converted to IFF format from Mimetics format, the loop points and length of the top octave are used to determine the values for the remaining octaves.

Multiple sampling is possible if you follow the rule that each octave should be half the size of the previous sample. The best way to create additional octaves is to use the option that translates one octave into another. This will create octaves that are exactly multiples of two. After you create the additional octaves, set the loop points for all ten octaves at once. Use the Sampler window to set the loop points rather than doing it individually while editing each samples. This way, they'll line up, as the IFF format wants.

Reference Section

This section addresses some of the more technical aspects of the Sampler software. The topics appear in alphabetic order.

Compression, Dynamics, and Amplitude Envelopes

Along with making the recording process a bit easier, compression can actually improve sound quality.

During the recording process, compression maintains a constant signal to noise ratio of 8 bits, or 48 db, which is just about full volume.

But while compression records the sound with a constant high signal to noise ratio, it levels out the dynamics, the fluctuations in volume.

But it's nice for strings to start quiet and then get loud, and for pianos to have a sharp attack, and to decay to a soft tone. Although compression levels out these dynamics, they can be reimposed with an amplitude envelope. The amplitude envelope provides an additional 6 bits of volume control that is used in addition to the 8 bit sample, providing a sound with 14 bits of range.

Experiment. Put percussive envelopes on strings, or slow attacks on drums.

Without compression, the recording can retain the dynamics of the original sound. Only the peak levels of the sound are recorded at full volume, the rest will be less.

Octaves

There are ten octaves to work with. Each octave is assigned a sample. When the note A is played, the sample plays back at the same rate it was recorded. All other notes in the octave, on up to G#, are created by playing the sample back at progressively higher speeds (the chipmunk effect.) The octave A is, of course, the next octave sample played at the original rate.

You may find that it is not necessary to record a sound into all ten octaves. Octaves 2 and 3 are usually enough for a bass, for example. So, you could put the bass on 2 and 3, a guitar on 4 and 5, and various sound effects on 6, 7, and 8. Or, you can put an entire ten piece drum kit together, one drum for each octave. Now, each type of sound is not only accessible, but can be tuned over an octave range. Remember, there is no law that says

low octaves have to have low samples. If you favor your left hand, you can put the bass on octaves 8 and 9, and the piccolo on octaves 2, 3, and 4.

Each octave is from A to G#. The console keyboard supports the upper half of octave four, all of octave five, and the lower half of octave six. These and all the other octaves are supported by MIDI.

Recording

When the screen is red, the sampler volume level is being set. If the mode is noncompressed, the sampler hardware is given a specific level. If the mode is compressed, the level is determined by both the volume and the incoming signal. In this case, a volume of 10 and a very loud signal sets the recording level very low. A volume of 99 and a quiet signal sets a high recording level.

Sample Rates and Aliasing Distortion

There is a difference in the quality of sample recorded at the high and at the low sample rates. The high recording rate of 14080 Hz yields the highest quality sample. The low recording rate of 7040 Hz can produce aliasing distortion.

Why the difference in quality between high and low sample rates?

When a sampled sound is played, the sample rate combines with the sample to produce the original signal (which we want) plus an additional signal that is the sampling frequency minus the original signal's frequency. This is called aliasing distortion. For example, if a 4khz tone is played at the sample rate of 15 khz, the reproduced signal will have two tones - one at 4 khz, the second at 11 khz. To deal with this, the Amiga's sound producing circuitry has an output filter that effectively cuts off all signals above 7 khz. So, the tone of our example would be played back while the annoying 11 khz is removed by the filter.

By recording and playing back at a sample rate that is at least twice the filter cutoff frequency, we guarantee that no aliasing produced tones can make it through. The high sample rate, 14080 hz, does just that. The sound is recorded by the sampler at 14080 hz. The playback software treats this as the bottom of the octave, though, so the other notes of the octave are played back at progressively higher rates, reaching 26579 hz for G#.

At the low sample rate, 7040 hz, aliasing is a problem. For example, a 2 khz signal will result in two tones, one at 2 khz, the second at 5 khz.

To get a better feel for what is going on, try this:

Set the sample rate to low and record a whistle that bends down in pitch. When played back, you should hear a metallic tone that moves up in pitch while the original whistle moves down. Play a higher note (A is the bottom of the octave). The alias tone will also be higher and might possibly disappear. Play G#. The tone should be just about gone since a sample rate of 13290 hz is used for G#.

Aliasing distortion is a two ended problem. It occurs both at recording and playback time. The sampler has a filter built in to deal with this. However, we should still have additional problems at the low rate, because the aliasing distortion should slip under the filter and end up being recorded in with the sample. The sampler avoids this by always recording at 14080 hz. At the low sample rate, the signal is run through a software filter that halves the data rate, while avoiding folding the sample rate and accompanying distortion into the data.

By the way, the sampling rate of 14080 hz is no magic number. It is A440 times 32, which is a standard for Amiga programs that generate musical sound. All other notes of the octave are obtained by playing back at progressively higher rates.

Index