

# **User manual and reference**

© Insigma Technologies Ltd. 1996-1999. All rights reserved.

Insigma Technologies Ltd  
PO Box 60  
Cirencester  
Gloucestershire  
GL7 5ZE  
UK

All rights reserved. No part of this publication may be reproduced in any form or by any means (graphic, electronic or mechanical, including photocopying, recording, taping or information storage and retrieval systems) without the prior permission in writing of Insigma Technologies Ltd.

H6, Hyperceive and the Hyperceive 'globe' logo are trade marks of Insigma Technologies Ltd. Java™ and the Java™ coffee cup logo are trade marks of Sun Microsystems Inc. Windows, Windows NT and Internet Explorer are trade marks of Microsoft Corporation Inc. Cool Edit is a trade mark of Syntrillium Software Corporation. Pentium is a trade mark of Intel Corporation. Netscape is a trade mark of Netscape Communications Corporation.

To the extent permitted by law, this user manual is provided 'as is' without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of satisfactory quality, fitness for a particular purpose, or non-infringement. The software described in this document is furnished under a licence agreement, set out in the following pages. The software may be used only in accordance with the terms of that license agreement. It is against the law to copy, distribute or use the software except as specifically allowed in the license.

This publication could include typographical errors. Changes are periodically added to the information herein; these changes will be incorporated in new additions of the publication. Insigma Technologies Ltd may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

# **SOFTWARE LICENCE**

**READ THIS LICENCE CAREFULLY BEFORE YOU INSTALL 'HYPERCEIVE H6' (THE SOFTWARE). BY CONTINUING THE INSTALLATION OF THE SOFTWARE, BY LOADING OR RUNNING THE PROGRAM, OR BY PLACING OR COPYING THE PROGRAM ONTO YOUR COMPUTER HARD DRIVE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT CLICK ON THE 'NO' BUTTON AND DESTROY THE ORIGINAL DISK AND OR DELETE ALL COPIES OF THE PROGRAM.**

**COMMERCIAL USE IS PROHIBITED: UNDER NO CIRCUMSTANCES SHALL YOU, THE END USER, BE PERMITTED, ALLOWED OR AUTHORISED TO COMMERCIALY EXPLOIT THE SOFTWARE.**

This software is supplied free of charge for the purpose of demonstrating the potential of the other products in the Hyperceive range from Insignia Technologies. If you wish to use the product commercially then you must purchase a copy of either Hyperceive or Hyperceive Professional.

## **Licence**

This Licence permits you to:

- a use the Software by Insignia Technologies contained on this CD ROM on more than one computer;
- b transfer the Software from one computer to another,
- c use no more than one 6 track arrangement at a time. Consecutive usage of two or more H6 arrangements within a Web site or Web page is strictly forbidden.

If you fail to comply with any terms of this Agreement, YOUR LICENCE IS AUTOMATICALLY TERMINATED. In that case, you must cease using the Software and destroy all copies of the Software then in your possession or control.

**You must obtain the consent of the relevant copyright owner or appropriate licensing body before using the Software to reproduce, adapt, distribute, play or perform copyright works belonging to third parties. Copyright works include literary, dramatic and musical works, films and sound recordings. Insignia Technologies will not be liable for any loss or claim resulting from any failure by you to obtain such consent and you shall indemnify Insignia Technologies for any claim, liability or expense resulting from any infringement of copyright by you.**

## **Copyright**

Save as provided below, Insignia Technologies owns the copyright and all other

intellectual property rights in the Software. Except as expressly permitted by this Licence or otherwise expressly permitted by law, you may not copy, modify, adapt, merge, translate, reverse engineer, decompile or dis-assemble the Software or any part of it.

### **Use of sample sound files and demonstrations provided on the Software installation CD**

Insignia Technologies owns the copyright in the sample sound files and demonstrations included in the Software package. You may install the sample sound files on your internet web site, but you may not use them for any other purpose or do any other act restricted by copyright in respect of them without obtaining written consent from Insignia Technologies. You may use the demonstrations only for purposes of reference but you may not use them for any other purpose or do any other act restricted by copyright in respect of them without obtaining written consent from Insignia Technologies.

### **Dealing with the Software**

You must treat the Software like any other copyright material, except that you may make copies of the Software to give to other persons only in its compressed and un-registered format. You may not charge or receive any consideration from any other person for the receipt or use of the Software. You agree to ensure that any user of the Software licensed hereunder complies with this Agreement.

### **Warranty**

Insignia Technologies does not warrant that the Software will meet your particular requirements or that the operation of the Software will be uninterrupted or error-free. Any other warranties, terms and conditions implied by statute, common law or otherwise in respect of the Software are excluded to the fullest extent permitted by law.

**TO THE FULLEST EXTENT PERMITTED BY LAW, INSIGMA TECHNOLOGIES' TOTAL LIABILITY IN RESPECT OF THE SOFTWARE SHALL IN NO CIRCUMSTANCES EXCEED THE PRICE PAID FOR THE SOFTWARE PACKAGE.**

**TO THE FULLEST EXTENT PERMITTED BY LAW, INSIGMA TECHNOLOGIES EXCLUDES ITS LIABILITY FOR LOSS OF PROFITS, LOSS OF BUSINESS INFORMATION, LOSS OF REVENUE OR GOODWILL, BUSINESS INTERRUPTION, LOSS ARISING FROM THIRD PARTY CLAIMS AND ANY OTHER INDIRECT OR CONSEQUENTIAL LOSS, WHETHER ARISING IN CONTRACT, TORT (INCLUDING NEGLIGENCE) OR OTHERWISE.**

Nothing in this Licence restricts Insignia Technologies' liability for death, personal injury or direct physical damage to tangible property caused by its negligence. To the extent that they apply to this licence, nothing in this Licence affects the statutory rights of consumers under English law.

YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, YOU UNDERSTAND THIS AGREEMENT, AND UNDERSTAND THAT BY CONTINUING THE INSTALLATION OF THE SOFTWARE, BY LOADING OR RUNNING THE SOFTWARE, OR BY PLACING OR COPYING THE SOFTWARE ONTO YOUR COMPUTER HARD DRIVE, YOU AGREE TO BE BOUND BY THIS AGREEMENT'S TERMS AND CONDITIONS.

YOU FURTHER AGREE THAT, EXCEPT FOR WRITTEN SEPARATE AGREEMENTS BETWEEN INSIGMA TECHNOLOGIES AND YOU, THIS AGREEMENT IS A COMPLETE AND EXCLUSIVE STATEMENT OF THE RIGHTS AND LIABILITIES OF THE PARTIES.

THIS AGREEMENT SUPERSEDES ALL PRIOR ORAL AGREEMENTS, PROPOSALS OR UNDERSTANDINGS, AND ANY OTHER COMMUNICATIONS BETWEEN VC AND YOU RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

Insignia Technologies does not offer support for this product. However, the Web site <http://www.hyperceive.com> hosts regularly upgraded information that may be of relevance and use when utilising the software.

Please contact Insignia Technologies at

Insignia Technologies Ltd  
PO Box 60  
Cirencester  
Gloucestershire  
GL7 5ZE  
UK

if you have any questions relating to this Licence.

This Licence is governed by English law.

The Software incorporates Java™ software belonging to Sun Microsystems Inc. (Sun). The following Sun licence terms govern the use of Java™ software:

## **1 Restrictions**

Java™ software is confidential copyrighted information of Sun and title to all copies is retained by Sun and/or its licensors. You shall not decompile, disassemble, decrypt, extract, or otherwise reverse engineer Java™ software. Java™ software may not be leased, assigned, or sublicensed, in whole or in part. Java™ software is not designed or intended for use in on-line control of aircraft, air traffic, aircraft navigation or aircraft communications; or in the design, construction, operation or maintenance of any nuclear facility. You warrant that you will not use or redistribute the Java™ software for such purposes.

## **2 Trademarks and Logos**

This Agreement does not authorise you to use any Sun name, trademark or logo. You acknowledge that Sun owns the Java™ trademark and all Java™-related trademarks, logos and icons including the Coffee Cup and Duke (Java™ Marks) and agree to:

- i comply with the Java™ Trademark Guidelines at <http://java.sun.com/trademarks.html>
- ii not do anything harmful to or inconsistent with Sun's rights in the Java™ Marks; and
- iii assist Sun in protecting those rights, including assigning to Sun any rights acquired by you in any Java™ Mark.

## **3 Disclaimer of Warranty**

Java™ software is provided AS IS, without a warranty of any kind. ALL EXPRESS OR IMPLIED REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED.

## **4 Limitation of Liability**

SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY YOU OR ANY THIRD PARTY AS A RESULT OF USING OR DISTRIBUTING JAVA SOFTWARE. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE JAVA SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## **5 Export Regulations**

Java™ software, including technical data, is subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. You agree to comply strictly with all such regulations and acknowledge that you have the responsibility to obtain licenses to export, re-export, or import Java™ software. Java™ software may not be downloaded, or otherwise exported or re-exported (i) into, or to a national or resident of, Cuba, Iraq, Iran, North Korea, Libya, Sudan, Syria or any country to which the U.S. has embargoed goods; or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nations or the U.S. Commerce Department's Table of Denial Orders.

## **6 Restricted Rights**

Use, duplication or disclosure by the United States government is subject to the restrictions as set forth in the Rights in Technical Data and Computer Software Clauses in DFARS 252.227-7013(c) (1) (ii) and FAR 52.227-19(c) (2) as applicable.

<b>1 Introduction</b>	<b>9</b>
1.1 Product overview	10
1.1.1 What is H6?	10
1.1.2 Audio on the Internet and the H6 solution	10
1.1.3 How does H6 work?	10
1.1.4 How easy is H6 to use?	10
1.2 System requirements	11
1.3 About this guide	12
1.4 Before you start - important!	12
<b>2 Getting started</b>	<b>13</b>
2.1 Introduction	14
2.2 The player	14
2.3 Inserting your tracks	14
2.4 Adjusting track options	15
2.5 Arranging your tracks	17
2.5.1 Loopback points	19
2.5.2 Beat Length events	19
2.5.3 Unconditional jumps	19
2.6 Adjusting track playback	20
2.6.1 Play	20
2.6.2 Play Once	20
2.6.3 Needed by	20
2.7 Modem simulation	22
2.8 Saving and publishing	22
2.9 Incorporating your work into your site	24
<b>3 The editor in detail</b>	<b>25</b>
3.1 Introduction	26
3.1.1 The toolbar	26
3.1.2 The info bar	27
3.2 Editor menus	27
3.2.1 The File menu	27
3.2.2 The Edit menu	29
3.2.3 The Play menu	29
3.2.4 The Window menu	29
3.3 The track list	30
3.3.1 The track list menu	30
3.4 Track properties	32
3.5 The event list	32
3.5.1 The System Track	33
3.5.2 Other tracks	34
3.6 Editing	36
3.6.1 The editing context menu	36
3.6.2 Event-specific menu options	37
3.6.3 Event properties	37
3.7 Testing	38
3.7.1 The Play menu	38
3.7.2 Modem simulation	38
3.8 Publishing	39
3.9 Preferences	40
3.9.1 General preferences	40
3.9.2 Save preferences	40
3.9.3 Publish preferences	41

3.10 File layout	41
3.11 Applet parameter reference	43
3.11.1 Audiobase	43
3.11.2 Songfile	43
3.11.3 Colour	43
<b>4 Tutorial</b>	<b>44</b>
4.1 Preparing your sound files	45
4.1.1 Recording sounds	45
4.1.2 Saving sounds	45
4.1.3 Editing sounds	46
4.2 Arranging your sound files	50
4.2.1 Inserting files	50
4.3 Publishing	51
4.4 Orchestrating your site	52
<b>Glossary</b>	<b>54</b>



# 1 Introduction

## **1.1 Product overview**

### **1.1.1 What is H6?**

**H6** is an audio sequencing package that allows you to build soundtracks for use in Web sites. It takes small sound files and arranges them into sequences that play back either as continuous sound or specifically timed audio ideas. Any type of sound can be used - music, voice, sound effects. Once a sequence has been completed, **H6** will automatically create an HTML document that contains the information needed to make the sound run in any Web page.

Using **H6**, you can make sound run throughout a site, regardless of what the visitor is doing. Alternatively, the sound can be made specific to each page within a site, with the sound reflecting the visual content of the page. Thirdly, a combination of these approaches can be used to provide a continuous backing track throughout the site, with specific sounds attached to specific pages.

### **1.1.2 Audio on the Internet and the H6 solution**

It is an accepted problem - how to include audio on a Web site when bandwidth limits mean you really shouldn't be doing it. Well now you can and in a powerful new way with complete control at the click of a mouse.

Hyperceive products manage audio in a unique way:

- They use small snippets of audio which download quickly.
- They don't require any plug-ins or add-ons at the user end.
- They intelligently manage the flow of the audio files to create continuous audio where needed.
- They allow for specific timing of audio playback.
- They produce a complete HTML file.

### **1.1.3 How does H6 work online?**

Once a **H6** sequence file has been created, the user adds the following things to the directory for their Web page:

- the completed .hyp sequence file
- the directory with the sounds that have been used
- the Java™ 'engine' files which are automatically created by H6, and make the system work.

When someone visits the page, the .hyp sequence file is automatically downloaded together with the player file, which is a Java™ class file, and sets

the sequence playing. These files download very quickly, without disrupting the rest of the site. The player will then begin to download the audio files used in the sequence. Once the first audio file has downloaded, it will begin to play according to the sequencing information contained in the .hyp file. Meanwhile the player will be continuously downloading the other audio files. This gives continuous sound if required. In the event that a file hasn't downloaded by the time it is needed in the sequence, the player can be made to loop back to a particular point in the sequence and replay particular files until the next file is ready. Once each file has downloaded, the player will read the files from the cache memory of the user's computer, so that the files needn't be downloaded each time. This also means that once a **H6**-enhanced site has been looked at, it can be viewed again offline and the sound will still play.

### 1.1.4 How easy is H6 to use?

**H6** has been designed to be as user-friendly and intuitive as possible. It has a clear user interface and simple but powerful function controls. All the files required to make your sequence work on a Web page are automatically produced. The basics of sound file preparation are easy to master and the rest is down to your imagination.

## 1.2 System requirements

The basic system requirements for operating **H6** are:

- Windows 95/98 or Windows NT 4.0
- A PC (Pentium 120 or better, 200 recommended)
- 16MB RAM (32MB recommended)
- 10MB free hard disk space
- A multimedia sound card (games card or better)
- PC speakers
- Modem (optional)
- CD ROM drive

To view your work, you will need:

- A Java™-enabled Web browser
- Connection to the Internet

To edit your sound files, you will need:

- An audio editing package, such as Cool Edit from the Syntrillium Software Corporation. Cool Edit is readily-available shareware, and you can find a copy on your **H6** installation CD. Please remember to register your copy of Cool Edit with Syntrillium and to pay the shareware fee. You will then have access to the full range of Cool Edit's audio editing capabilities.

### 1.3 About this guide

This guide is divided into 6 sections:

- 1 This Introduction
- 2 Getting started - a quick start guide to get you going with H6.
- 3 The editor in detail - a detailed analysis of H6 and its functions.
- 4 Tutorial: an example project - a step-by-step guide through a Web site project which includes the preparation of audio files for use in H6.
- 5 Glossary - definitions of terms used throughout this guide.

### 1.4 Before you start - important!

According to the **H6** Software Licence, this software is supplied free of charge for the purpose of demonstrating the potential of the other products in the Hyperceive range from Insignia Technologies. If you wish to use the product commercially then you must purchase a copy of either Hyperceive or Hyperceive Professional. It is also forbidden to use more than one **H6** sequence file consecutively in one Web site. See the Software Licence (printed at the beginning of this document) for more details. Visit **[www.hyperceive.com](http://www.hyperceive.com)** for more information on the full version of Hyperceive and Hyperceive Pro.

**H6** is a limited version of Hyperceive. As you use **H6**, you will notice several features from Hyperceive that have been disabled. These include:

**6 tracks instead of 24**

**Modem simulation disabled**

**Jump events disabled**

**Only one beat length event allowed**

**Adjustment of default audio directory disabled**

These features afford far greater functionality and can be found in the full version of Hyperceive. For more details about their use, please see the main part of this manual.

## **2 Getting started**

## 2.1 Introduction

This overview is designed to take you through the basic functions of **H6**. To start you will need to:

- 1 successfully install your copy of **H6**
- 2 have your audio files prepared as 8 kHz, 8 bit mono Next/Sun **.au** files . There are some example files included in the installation.
- 3 place the audio files in a specific directory
- 4 open **H6**.

## 2.2 The player

The basic player functions are similar to those found on a CD player. The figure (*right*) shows from left to right:



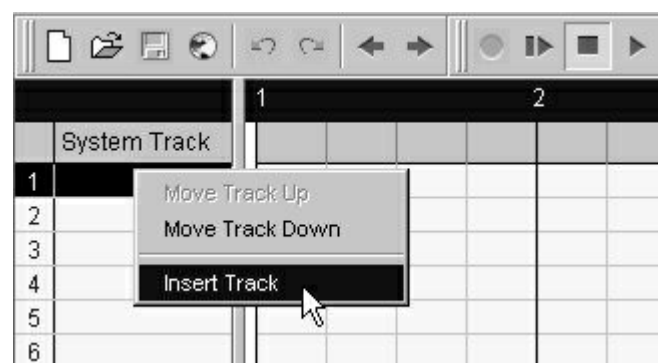
- Scroll to start of sequence
- Scroll to end of sequence
- Freeze playback to this point
- Play the sequence from a selected point
- Stop the playback
- Play the sequence from the beginning
- Pause the playback
- Rewind the sequence
- Fast forward the sequence

## 2.3 Inserting your tracks

Open the **File** menu on the menu bar and select **New** to create a new song or click the **New Song** icon.



Click the right mouse button in the box beside the number 1 under **System Track**. Select **Insert Track** from the menu. This will open a dialog box from which you can select the desired file from the directory in which you have saved your **.au** files. Highlight the first file for selection and click on **Open**.



## 2.4 Adjusting track options

The file selected (CLASSICAL1.au) appears in box 1 of the **Track list**, and is now ready to be inserted into the sequencing space. Double click on the beat position (the space between the vertical lines) where you wish to place your first audio file. A blue line appears, representing the complete file. In the screenshot, the file has been inserted into beats 1 and 3.



Now simply hit the **Play** icon and listen!

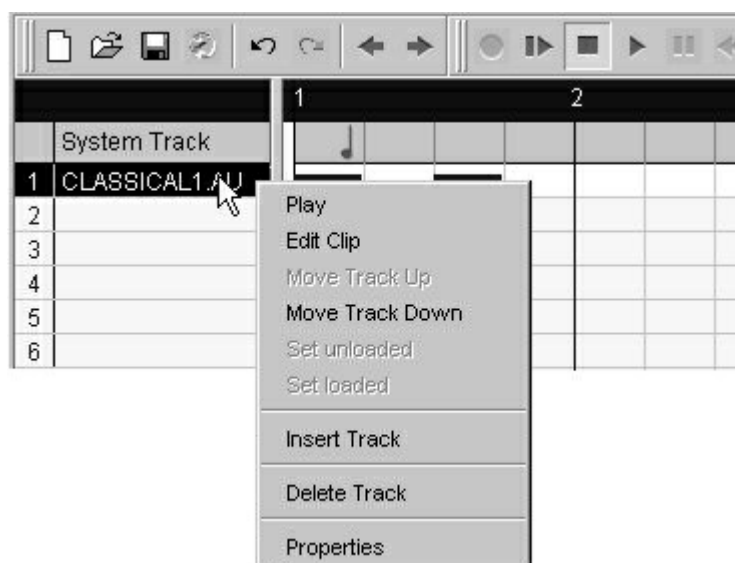


As you will see, you can place this file anywhere along its sequencing space that you wish. You are now ready to add your second file. Click on the right mouse button immediately below the first track and you can choose a second file from your audio directory. Add your files in the order in which they will be played.

## 2.4 Adjusting track options

By clicking the right mouse button on the audio track name, various options are available.

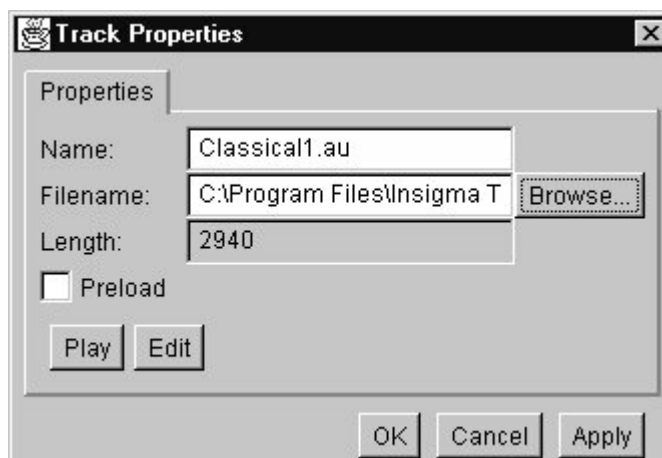
**Edit Clip** will automatically load this sound file into the audio editing software of your choice, assigned in the **Preferences** section within the **File** menu (*see next page*). This can also be activated by double clicking on the audio track name, allowing adjustments of the file to be made. Don't forget to save your file when you have finished editing it.





In this case, **Cool Edit** has been selected as the audio editing application. It is shareware and can be found on your **H6** installation CD, but please register your copy of **Cool Edit** with Syntrillium.

**Track Properties** allows you to rename your audio file or to replace it with an alternative file, selected through the Browse option:



You can give your file a nickname which corresponds to its sound content for easy recognition in the track list. This will not affect the file name.

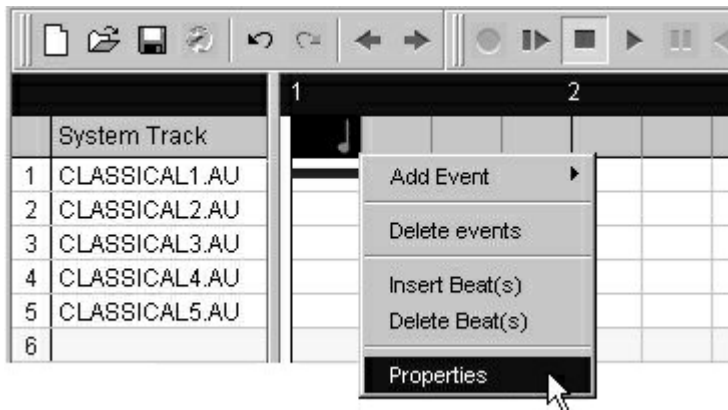
The remaining track options from the right-click selection are self-explanatory. A full list and explanations can be found in Chapter 3.



## 2.5 Arranging your tracks

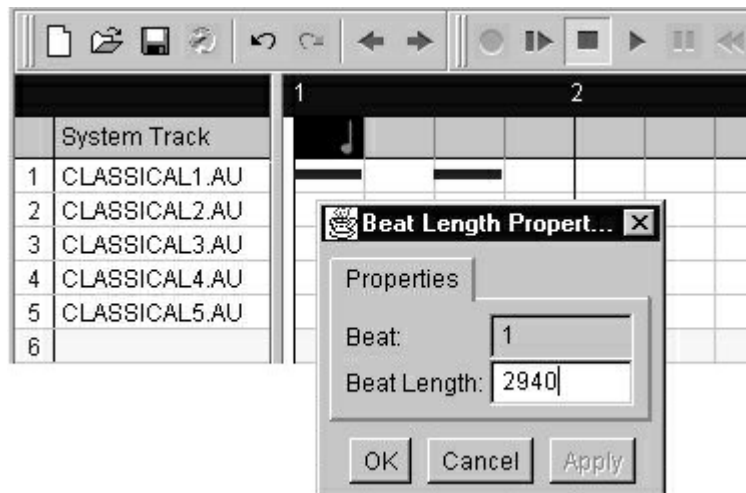
Now let's look at the features of **H6** which give control and variation to your sequence.

**H6** will play back your sequence at the playback speed set in the properties of a **Beat Length**. A musical note situated in the **System Track** at the top of the sequencing window shows that a beat length has been set.



To find out the value of the beat length, access the **Beat Length Properties** dialog by right-clicking the musical note...

...and selecting **Properties** from the drop-down menu.

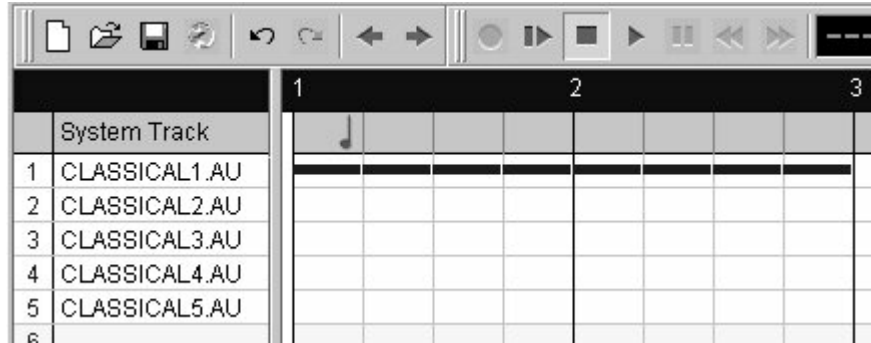


A beat length is a time value in milliseconds. **H6** will initially adopt a value that corresponds to the length in milliseconds of the **first** file to be inserted. The beat length dictates the speed at which the editor plays the sequence. The space between the vertical lines in the main sequencing window equalling one beat length. Therefore in the case shown, the first track has a beat length of 2940 milliseconds, and the space between each vertical line in the main sequencing window represents 2940 milliseconds.

The value in milliseconds of the beat length can be adjusted by changing the value in the **Beat Length Properties** dialog.

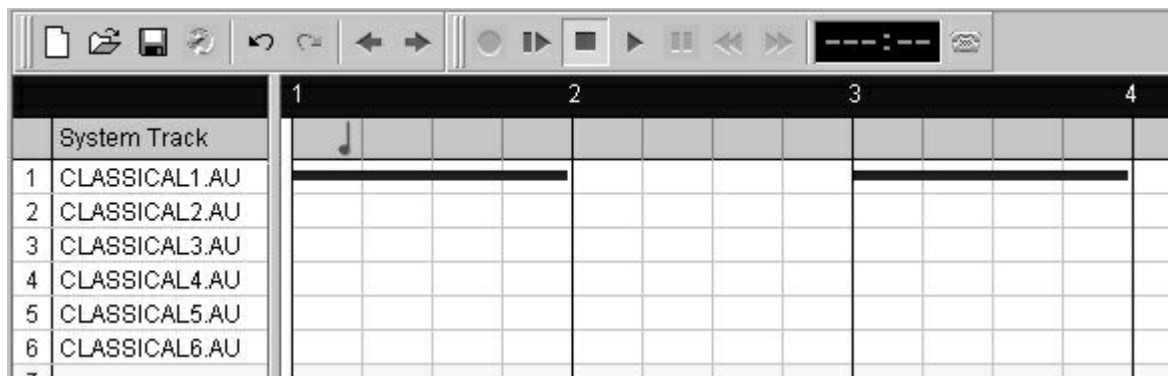
## 2.5 Arranging your tracks

As an example of choosing an appropriate beat length, place a succession of instances of the same inserted audio file to create a back to back sequence (*below*). If the beat length is correct, the files will play smoothly from one to the next, giving the impression of looping.



The correct time value to produce a smooth loop is normally slightly less than the actual length in milliseconds of the file. This allows each instance of the file to flow into the next without jumping. In the case of the CLASSICAL1.AU file shown, the beat length was reduced from 2940ms to 2400ms.

Once you have arrived at the correct time value, you can create time fractions by reducing the beat length value by a particular amount, say 1/2 or 1/4.



In the example shown above, the time value of the beat length has been reduced from 2940ms, as seen on the previous page, to 735ms, which is a quarter of the original value. This means that each beat is now a quarter of the length of the file, so the sound file stretches over four beat spaces rather than just one. Now other files can be placed in an off-beat sequence relative to CLASSICAL1.AU. This allows you to control more accurately how tracks are synchronised with one another.

**Note:** having adjusted the beat length, you may have to reinsert your files to fit the new beat sequence; otherwise they may overlap.

Further events can be added to occur at certain times throughout your sequence. A list of the available events pops up on a click of the right mouse button at the place of your choice along the shaded bar of the **System Track**.

Each event is represented by its own icon. The available events are:

### 2.5.1 Loopback Points

A loopback point can be placed anywhere in the sequence that represents a suitable point to return to in the event of the next file not having downloaded to the end user's machine in time. Sensible placement of loopback points enables a continuous flow of sound should it be required.

### 2.5.2 Beat Length Events

The beat length dictates the speed at which the editor plays the sequence.

**Note:** only one beat length event (at the beginning of a sequence) is allowed in **H6**. The full version of Hyperceive allows unlimited beat length events, and therefore far greater control over the timing and synchronisation of the files.

### 2.5.3 Unconditional Jumps

Unconditional Jump events are not available in **H6**. In the full version of Hyperceive, they are repeat points separated by up and down arrows. These arrows can be placed wherever you wish a section to repeat in a continuous loop.

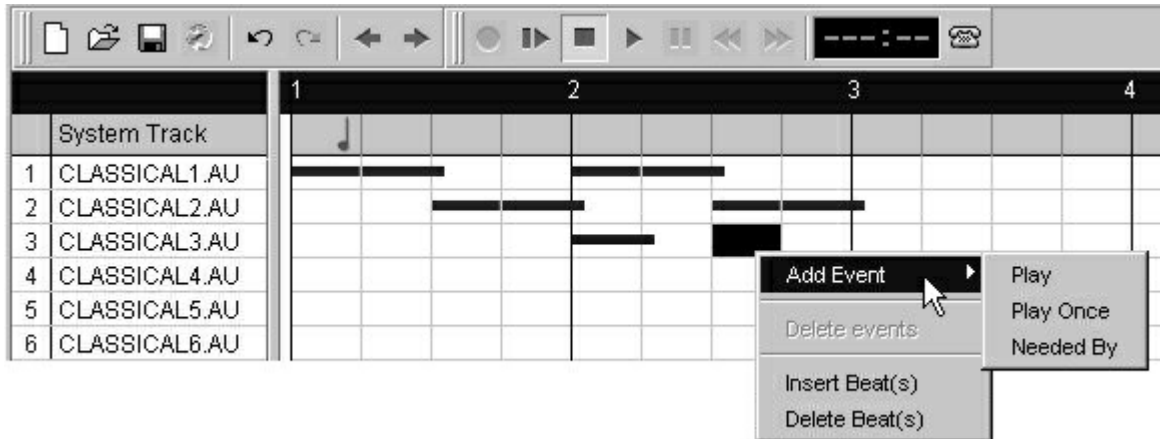
Also triggered by the right click are the Insert beats and Delete beats options. These provide a facility for lengthening or shortening your sequence. The number of beats is entered in the box, as seen in Fig 2.15. The new beats are inserted before the selected beat.

**Delete beats** will delete all selected beats.

## 2.6 Adjusting track playback

You've seen how the horizontal blue bars represent the audio files of your sequence. Now let's look at some of the functions that effect this layout.

We have already inserted tracks in the sequence by double clicking within the desired beat space (section 2.3). The figure below shows another way of achieving this.



A click on the right mouse button within a beat space will reveal a pull-down box offering the following alternatives:

- 1 Play
- 2 Play Once
- 3 Needed By

### 2.6.1 Play

**Play** inserts an instance of a file within the sequence in the same way as a double click.

### 2.6.2 Play Once

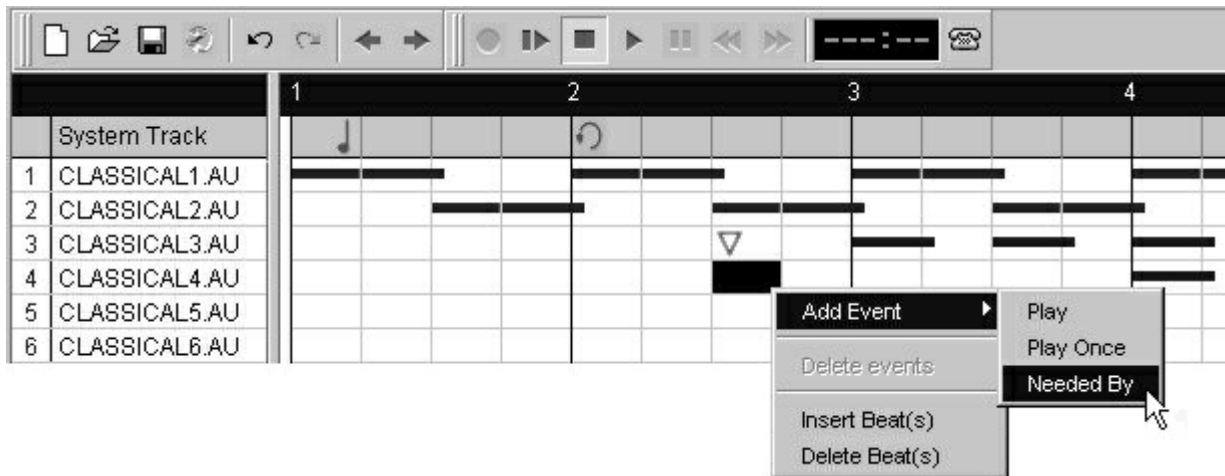
This option will insert a pale blue bar into the sequencing window to represent an audio file that will not be repeated in the event of a loopback. This is especially useful for voice files such as welcome messages which would sound unusual if repeated.

### 2.6.3 Needed By

By placing the **Needed by** icon ahead of an audio file position, you can instruct the player to return from the **Needed by** triangle to the last loopback point until any files flagged as **Needed by** have downloaded. This is especially useful for guaranteeing that audio files will play together, and can be likened to pre-fetching audio files.

## 2.6 Adjusting track playback

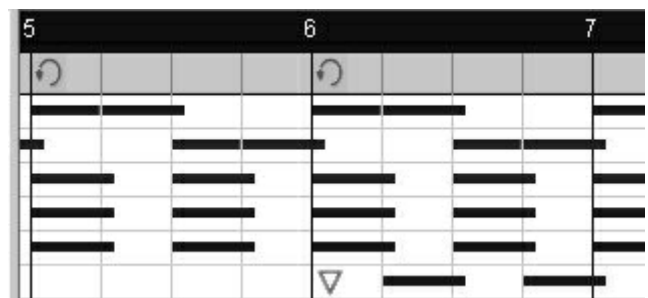
The figure below shows **Needed by** placements for CLASSICAL3 and the insertion of another **Needed by** for CLASSICAL4.



Note that the **Needed by** triangles are at the same beat position (beat 3 of bar 2). This is telling the player that both files must have downloaded and be ready by that point. If not, the player loops back from the **Needed by** triangles to the loopback point at beat 1 of bar 2 until both CLASSICAL3 and CLASSICAL4 have downloaded, allowing the two files to play together.

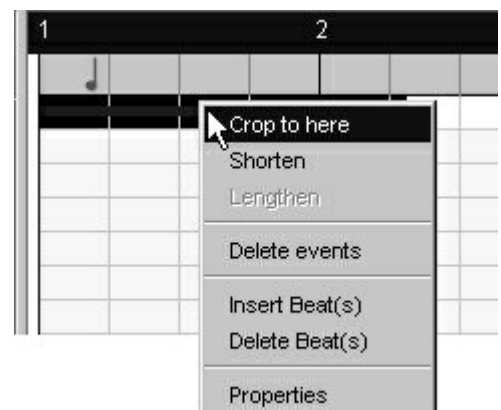
The **Needed by** option also enables the controlling of loopback to suitable points to retain the musical sense of an sequence, even over a slow Internet connection.

Placing a **Needed by** event as shown here will make the sequence jump back from beat 1 of bar 6 to the loopback point at beat 1 of bar 5. The composer has decided that this makes a more musically ideal loop sequence than to have the player jump back from beat 4 of bar 5. The player will only jump back if Track 6 has not downloaded by the time the **Needed by** triangle is reached.



By clicking with the right mouse button on an inserted audio play event, the drop down box shown (*right*) will appear.

Here we have the option to shorten a play event. Cropping the play event will reduce its length to the end of the selected beat. This can be used to tighten the sound of a file or to instruct the player to reproduce only part of a file such as the first two beats in a drum loop. Lengthen will undo the cropping.



## 2.7 Modem simulation

The modem simulation function is not available in **H6**. In the full version of Hyperceive, it is designed to imitate the download behaviour of a modem. It is particularly useful for testing the viability of your sequence and the correct placement of Loopback points and Needed by placements. This way you know how your sequence will sound under a variety of connection speeds without having to upload it to your server at each testing.

## 2.8 Saving and publishing

Before we look at the publishing function of **H6**, we shall briefly consider the everyday functions such as **Save**.



The fuigure shows from left to right:

- New song
- Open an existing song
- Save a song
- Publish a song
- Undo an action
- Redo an undone action
- Scroll to beginning of song
- Scroll to end of song

Throughout the process of building a song, it is essential to save your work frequently. The **H6** file format we use for saving the sequences has the extension **.hyp**.

On saving your work you will be asked whether you want to **copy** the audio files, or **reference** them. **Copy** makes sure that all the audio files you have used in your sequence are copied to your audio directory (creating the directory if it doesn't exist). **Reference** will not make copies of the files; **H6** will expect you to save your files to the right location so that they can be found later.

Finally, the time will come to publish your sequence as an HTML page, ready for putting onto the World Wide Web.

The first step is to decide on a name for the example HTML document which **H6** produces when you publish your sequence. Enter this name in the box marked **Leafname of example HTML file** in the **Publish Preferences** section (under **Preferences** in the **File** menu). The default name for the file is *Example.html*. You should also make sure that the two boxes marked **Save example HTML file** and **Always overwrite example HTML file** are ticked.

Now just hit the publish icon, agree to the overwriting of the .hyp file and you will publish an HTML page, complete with all your audio ingredients, ready for the Web.



```
<!-- Hyperceive playback system (c) Copyright Insignia Technologies Ltd. 1999 -->
<APPLET CODE="H6.class" ARCHIVE="player.jar" WIDTH=2 HEIGHT=2>
  <PARAM NAME="audiobase" VALUE="audio">
  <PARAM NAME="songfile" VALUE="Mysong.hyp">
  <PARAM NAME="colour" VALUE="0,0,0">
</APPLET>
```

Shown above is the section of the sample HTML page which contains all the relevant information required for the sequence to play. In this case the player will look for the audio files in a directory called *audio*, and the **H6** .hyp sequence is called *Mysong.hyp*. These names will change for each sequence you publish, depending on what names you have given your audio directory and **.hyp** file. Cut and paste this section into the page on which you wish the sound to appear, or use the sample page as the basis for a new page.

Other items you will require in conjunction with this page on your server are:

- The Audio directory, complete with the audio files from your sequence.
- The applet files, which will be copied to your local directory by the publishing option of the editor (ensure the Copy applet files option is ticked in the preferences section, see Fig 2.23). These files are:
  - H6.class
  - H6Base.class
  - H6Clip.class
  - H6Player.class
  - player.jar
- The **.hyp** file that you have created using **H6** (which contains the sequencing information).

Shown below is a folder with all the necessary files in place to play an sequence called *Mysong.hyp* on a Web page called *Mypage.html*.





## 2.9 Incorporating your work into your site

For the purposes of this overview, we shall look at the two most straightforward methods of incorporating your sequence into your Web site.

- 1 Placing the applet information into the HTML of individual pages.
- 2 Placing the applet information into its own HTML page to sit within a persisting frame of a frameset.

Page by page your Web site can deliver different audio messages. By creating a separate **.hyp** file for each page in your site, you can make the sound change from page to page.

Alternatively, it is often the case that a site will have continuous sound running throughout. To achieve this is slightly more complicated but the results are great, with audio content continuing regardless of where you may navigate through the site. In the full version of Hyperceive, the Jump event allows indefinite repetition of a sequence. This guarantees continuous audio in a site.

The HTML shown below is for a frameset containing a page called *sound.html*. This is the persisting frame that holds the applets and sequencing information, and it is sized to zero so that it is invisible and does not interfere with the layout of the frames.

```
<HTML>
<HEAD>
<TITLE>hyperceive demonstration pages</TITLE>
</HEAD>

<FRAMESET COLS="0,177,*" FRAMEBORDER="yes" BORDER="0" FRAMESPACING="0">
  <FRAME SRC="sound.html" SCROLLING="auto" RESIZE="no">
  <FRAME SRC="navbar.html" SCROLLING="no" RESIZE="no">
  <FRAME SRC="home.html" SCROLLING="auto" RESIZE="no" NAME="main">

</FRAMESET>
</HTML>
```

In this example, *navbar.html* would be the navigation or menu bar frame from which the site can be navigated. *home.html* would be the main introduction page (this would also be the frame where the other pages in the site would appear as they were called up via the navigation bar).

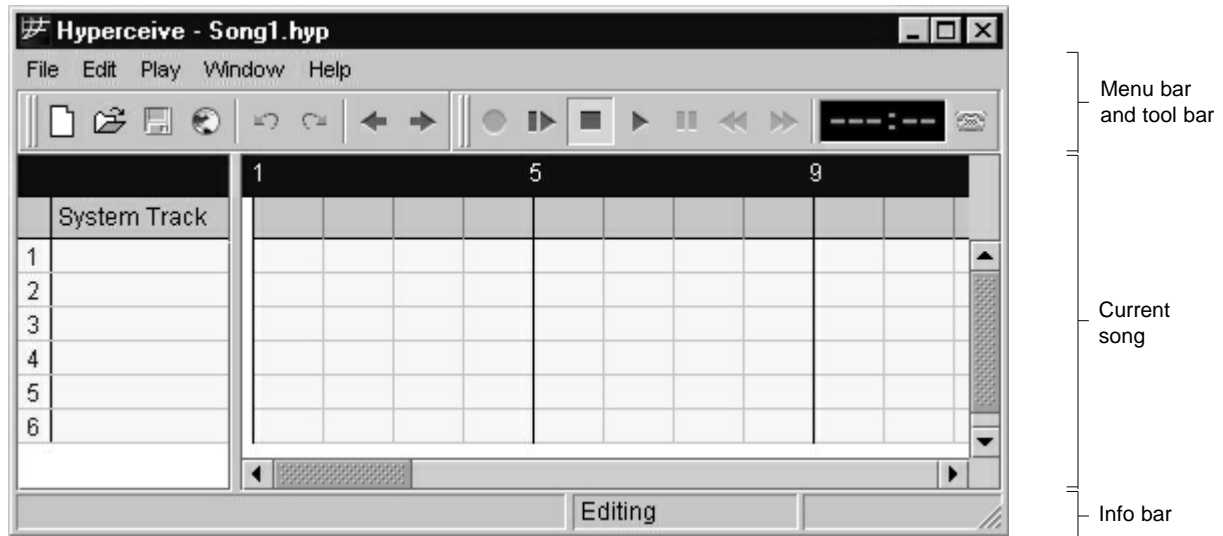
If *sound.html* holds the relevant applet tags for the desired audio content, the sound will continue throughout the site, regardless of which page is visited.



## **3 The editor in detail**

## 3.1 Introduction

The main editor window is broken into three sections: at the top is the menu bar and toolbar, at the bottom the info bar, and in the middle is an area, initially blank, where the current song is displayed.



### 3.1.1 The toolbar

At the top of the window, under the menu bar, is the editor toolbar.



The buttons, from left to right, are:

- New song
- Open song
- Save song
- Publish song
- Undo
- Redo
- Scroll to start
- Scroll to end
- Freeze
- Play from selection
- Stop
- Play
- Pause
- Rewind
- Fast forward
- Position counter
- Modem simulator (disabled)

### 3.1.2 The info bar



The **H6** info bar, at the bottom of the window, gives useful information about the current status of **H6**. At the far left is a description of the object that the mouse pointer is over; to the right is the current status of the editor (Editing, Loading, etc.), and at the far right is a progress bar, which shows progress during loading, saving and testing (playback).

## 3.2 Editor menus

### 3.2.1 The File menu

The **File** menu contains the following options:

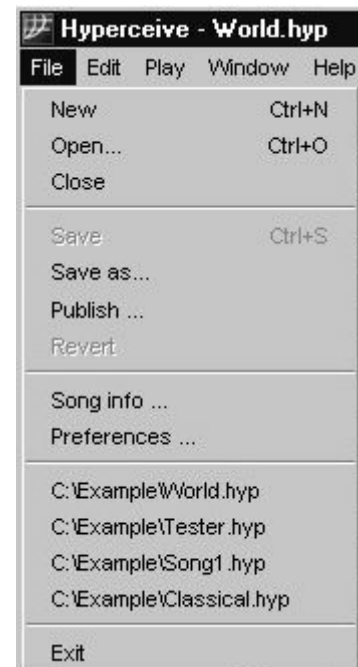
#### New

Creates a new song. This will be a blank sequence with 48 beats (twelve four beat bars).

#### Open

Brings up a file dialog, which allows you to find a **.hyp** song file and open it. **H6** files typically have the extension **.hyp**. If the song is already open and has been modified, you will be asked whether you want to revert to the saved file on disc, or not open the saved file.

When a file is loaded, the editor's title changes to **Hyperceive - filename.hyp**. If the file has been modified, the filename is followed by a star (\*). If the editor window is minimised, the title will change to **filename.hyp\* - Hyperceive**.



If an audio file used by the song cannot be found, you will be warned. You'll still be able to edit the song, but you won't be able to test it fully until you let the editor know where the audio file is. You can do this by reinserting the file in the track list via the **Track Properties** dialog (see section 3.4).

#### Close

Closes the current song, switching to the next if there are others open.

#### Save

Saves the current song. If the song has not previously been saved, or if the

previous save location is now invalid, then a file dialog will appear from which you choose where to save the song.

Due to the way **H6** songs are stored, it is necessary to save all audio files used by the song in a single directory where the editor and playback systems can find them later. If the files are not already present in that directory, you will be asked whether you want to copy them or reference them (see also section 3.9.5). The former choice copies the files to the Audio directory. This directory will be created if it doesn't already exist. Referencing the files simply means that the filenames alone are remembered, without copies being made.

You can control this feature by checking the box labelled **Ensure audio clips accessible on save** in the **Save** tab of the **Preferences** dialog.

### **Save as**

Brings up a file dialog, allowing you to save the current song in a different place, or with a different name.

### **Publish**

Brings up a file dialog, allowing you to publish the current song. See section 3.8, **Publishing**.

### **Revert**

Reverts to the last saved version of the file, discarding all the changes made since then.

### **Song info**

Brings up the song information dialog, containing the title, author and copyright notice for the current song. These are embedded in the **H6** song file when saved.

### **Preferences**

Brings up the preferences dialog. See section 3.9, **Preferences**.

### **Recent files**

**H6** remembers the last four files you edited, and you can quickly reload them by clicking on their filenames.

### **Exit**

Quits **H6**.

### 3.2.2 The Edit menu

The **Edit** menu contains the following options:

#### Undo

Undoes the last action. All editing actions are stored in memory, and may be undone and redone.

#### Redo

Redoes the last action.

#### Delete events

Deletes all selected events.

#### Select all

Selects the entire song.

#### Select none

Deselects all selected items.



### 3.2.3 The Play menu

The **Play** menu contains the following options:

**Play**, **Play from selection**, **Pause**, **Rewind**, **Fast forward**, **Stop**, **Freeze** - see section 3.7, **Testing**.

#### Scroll to start

Moves the view of the current song so you can see the first beat.

#### Scroll to end

Moves the view of the current song so you can see the last beat.



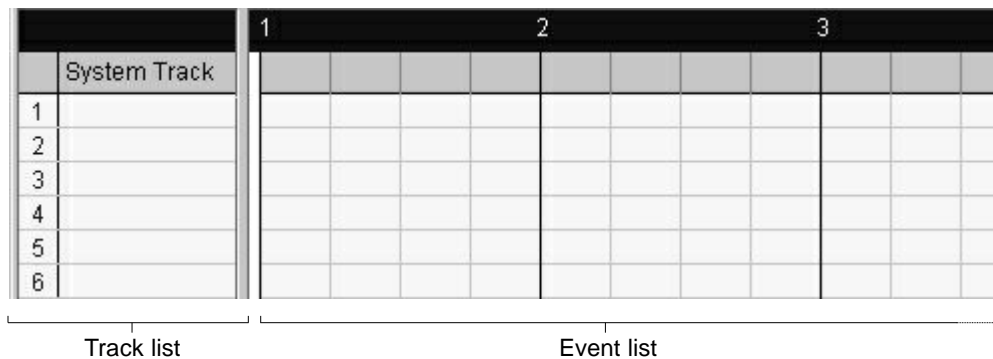
### 3.2.4 The Window menu

The **Window** menu contains a list of the songs that you have open at any time; the current song is shown with a tick. Selecting a different song will switch to that one.



## 3.3 The track list

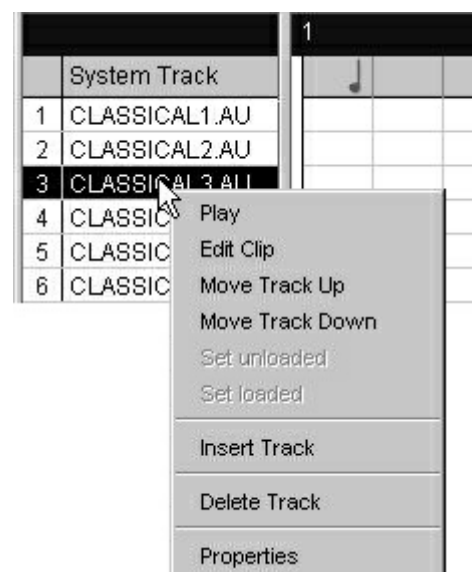
Once you have created a new song, or loaded a previously saved one, the central section of the editor will become active. It is split into two sections; on the left is the track list, and on the right is the **event list**, containing the sequencing information itself. This section deals with the track list, and the next, **The Event List**, deals with the song sequencing information.



If you have created a new song, the track list will initially only contain one track, labelled **System track**. This track is always at the top of the track list, runs throughout the sequence and cannot be deleted; it contains general commands to the **H6** player, such as 'set the beat length', or 'loopback to here'. (see section 3.5).

To select tracks in the track list, click with the left mouse button. You can select more than one track by holding down the **Ctrl** key while clicking on the tracks; you can select a range of tracks by holding down the **Shift** key as you click (Fig 3.12). You cannot select the system track.

Most actions, both in the track list and in the event list, are accessed through **context menus**. To bring up the context menu for the track list, click the right mouse button with the cursor on a particular track. A small popup menu will appear, containing some of the following options. Exactly which ones appear will depend on whether you have one or more tracks selected.



### 3.3.1 The track list menu

The Track List menu (*right*) contains the following options:

### **Play**

Plays the audio file associated with the selected track.

### **Edit clip**

Loads the audio file into an audio editor to allow you to edit it. Double clicking on the track name in the track list does the same.

The audio editor used may be configured in the **General** tab of the **Preferences** dialog. You can use whatever application you prefer (it must support Next/Sun **.au** files. Throughout this guide, we have used Cool Edit, which is easily available shareware and can be found on the **H6** installation CD. Please remember to register your copy of Cool Edit with Syntrillium Software Corporation).

If you edit a sound file using your audio editor and save it, **H6** has no way of knowing that the file has changed. In this event you need to tell it to refresh the information for that file; see section 3.4, **Track properties**, below.

### **Move track up, Move track down**

Moves the selected track up or down the track list relative to other tracks.

### **Set unloaded**

Sets the track as if it had not yet downloaded. A red cross appears over the track number.

### **Set loaded**

Undoes the **Set unloaded** command for the selected track.

### **Insert track**

Opens a file dialog for you to select an audio file to use for the new track. Next/Sun **.au** files usually have file extension **.au**, but they may be stored using a different extension, so the editor will let you select any file; it then checks that it's a valid audio file, warning you if it isn't. Please note that many Java™ implementations only accept audio files saved as 8kHz, 8 bit mono, mu-law **.au** files. The editor will warn you if you select one which plays at a different frequency. All sample audio clips supplied with H6 play at 8kHz (see also chapter 4).

### **Delete track(s)**

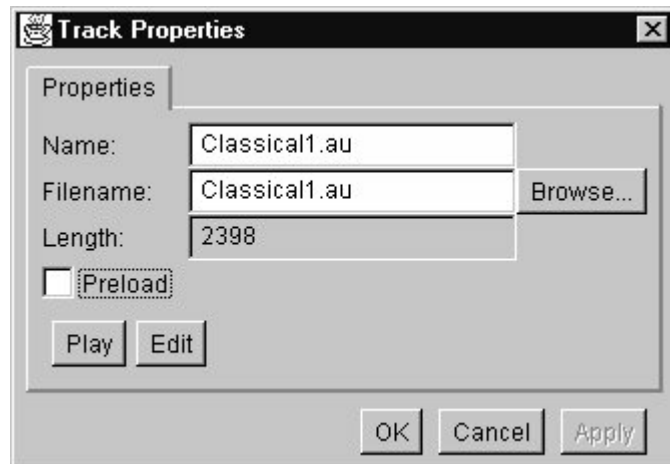
Deletes the selected tracks.

### **Properties**

Opens the **Track properties** dialog for the selected track.

## 3.4 Track properties

The **Track properties** dialog allows you to alter certain information about the selected track. At the top is the track name - this is the name used on the track list. Below this is the filename of the audio file associated with the track. You can change it either by typing the filename in directly, or by clicking on the **Browse** button to the right, which brings up a file dialog.



Below this is the length of time that the audio file will play for, in milliseconds. This is provided for information only.

At the bottom are two buttons, **Play** and **Edit**, which correspond to the **Track list** menu options described in section 3.3.1, above.

To refresh a track, open the **Track properties**, click on the **Browse** button, and click **Open** to reload the file, then **Okay** again to close the **Track properties** dialog. This selects the same file again, updating the track in the song with any changes made to the file in an audio editing package.

The **preload** check box allows you to set files preloaded. This will cause those files to download to the end user before the player starts playing. This ensures that the sound will start as soon as the first file has downloaded. Usually you would set the first file in the track list to preload. Occasionally you may need more than one file to preload if it is essential to make the first loopback settings of your sequence work properly. Files set to preload appear in blue in the track list.

## 3.5 The event list

		1 2 3											
System Track													
1													
2													
3													
4													
5													
6													

Track list                      Event list



The event list is numbered along the top, every bar (four beats). Each beat is separated by a vertical line from those on either side of it; each track has a row in the event list, which displays the events that happen in that track at each beat.

You can alter how beats are numbered from the **General** tab of the **Preferences** dialog. The second option down is **Use “bar n, beat m” (bar numbers only in beat track)**. With this option turned on, the numbers at the top of the event list will be numbered as bars (1, 2, 3, etc.), while with it turned off, they will be numbered as beats (1, 5, 9, etc.). This also affects how the position of the mouse pointer is displayed in the info bar (see section 3.9, **Preferences**).

Each track can contain a series of events, each tied to a particular beat, which tell the player what to do in that track when it reaches that beat.

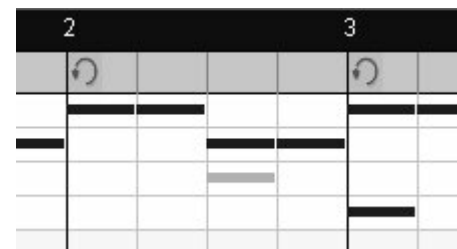
### 3.5.1 The System Track

The system track is a special track, because it is not associated with a particular audio file. Instead, as mentioned in the previous section, it contains general commands to **H6**. The possible commands are:

#### Loopback

A loopback point is used when **H6** finds that it needs a clip which is not yet available (typically because it has yet to download from the Internet). In this case, the player will return to the last loopback point that it passed. If it hasn't passed any loopback points, it will return to the beginning of the song.

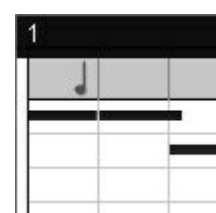
A loopback point is indicated by a light blue circular arrow. In this case, if the player reaches beat 4 of bar 2 and track 4 has not loaded, it will return to beat 1 of bar 2. It will continue to do this until track 4 has loaded. Track 3 here is a **Play once** event, so it will not play again on loopback (see section 3.5.2).



#### Beat length event

When **H6** plays back a song, it works out how fast to play it from the beat length, which is the number of milliseconds that each beat takes. The beat length defaults to the length of the first audio track inserted into the sequence. In **H6** only one beat length event can be used (at the beginning of the song). The full version of Hyperceive allows unlimited beat length events throughout the sequence.

A **Beat length event** is indicated by a light blue musical crotchet and can be adjusted to alter timing within the sequence and synchronisation of files (see section 2.5).

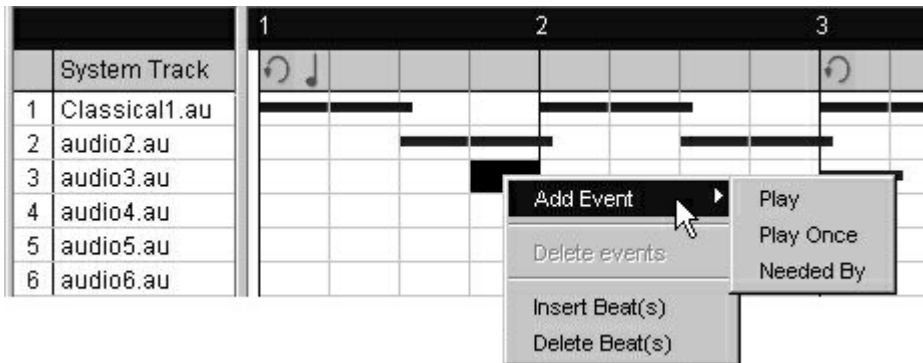


## Jump

The Jump event is only available in the full version of Hyperceive.

### 3.5.2 Other tracks

The events for a normal track, associated with an audio file, are accessed by clicking the right mouse button in the relevant position along the track for that file. They comprise:

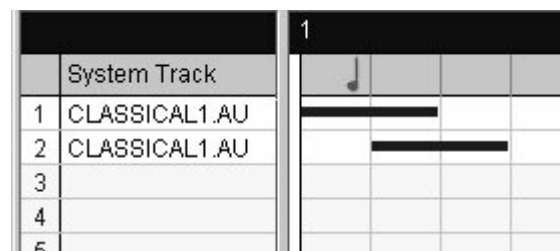


## Play

A **Play** event instructs **H6** to insert the audio file associated with this track in that particular beat of the track. This can also be done by double clicking in that beat.

Normally on playback, **H6** will keep playing an audio file until it gets to the end of that file, but it is possible to crop the playing to any particular beat (see section 3.6, **Editing**, below). Files are cropped automatically if the player finds another **Play** or **Play once** event. This means that if you want to have the same audio file being played concurrently at different points (for example, if you have an audio file which lasts two beats, and you want it to restart on every beat and to be able to hear the overlap), you will need to create two tracks associated with the same audio file(*right*).

A **Play** event is drawn as a dark blue bar. If it is truncated, the end of the bar is red to show where it is cut short.



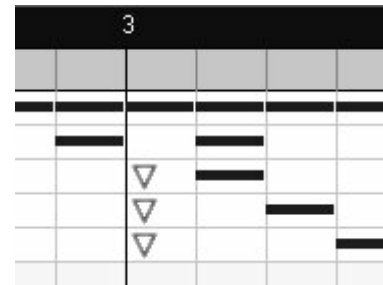
## Play once

A **Play once** event is exactly the same as a **Play** event, except it will only cause **H6** to play the audio file the first time it is encountered. This is useful for vocal samples, because if the player was forced to return to a **Loopback point** (see section 3.5.1), the repetition of spoken or sung phrases might confuse or annoy.

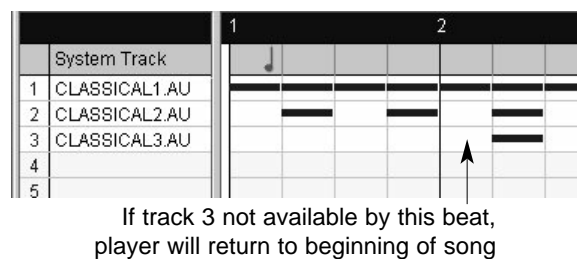
A **Play once** event is drawn as a light blue bar; truncation is indicated in the same way as for the **Play** event.

## Needed by

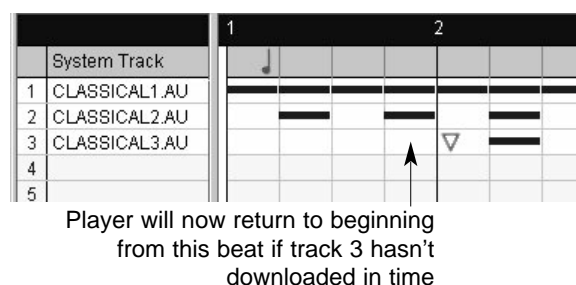
Normally **H6** can work out when an audio file is needed by looking for play events in its track, and then use that information to work out whether it needs to return to a loopback point. A **Needed by** event informs the player that this track must be downloaded by the relevant beat, without it starting to play then as well. In this way you have greater control over the position at which a the sequence will loop back if an audio file has not downloaded in time. The figure (*right*) shows 3 needed by events at the same beat.



Consider the following situation: you have a song in which the first two tracks start playing on the first and second beats of the first bar, and the third enters on the second beat of the second bar. If the audio file for the third track isn't available by the time the player reaches the second beat of the second bar, it will loop back to the beginning of the song - and the first bar will sound as if it lasted five beats:



Inserting a **Needed by** event in the third track on the first beat of the second bar will avoid this, since the player will loop back from the end of the first bar, instead of one beat later:



A **Needed by** event is indicated by a downward-pointing blue triangle.

## 3.6 Editing

Within the event list, a single click of the left mouse button deselects all events and selects only the one you clicked on; clicking with **Ctrl** held down adds to a selection, and clicking with **Shift** held down selects a range of events. You can also select events by dragging the mouse with the left mouse button held down; you can select all events in a beat by clicking in the area above the system track.

Some events, such as **Play**, can extend over more than one beat. If this is the case, the entire event will be selected.

With events selected, you can delete them using the **Delete events** option on the **Edit** menu. Alternatively you can delete events by double clicking on them.

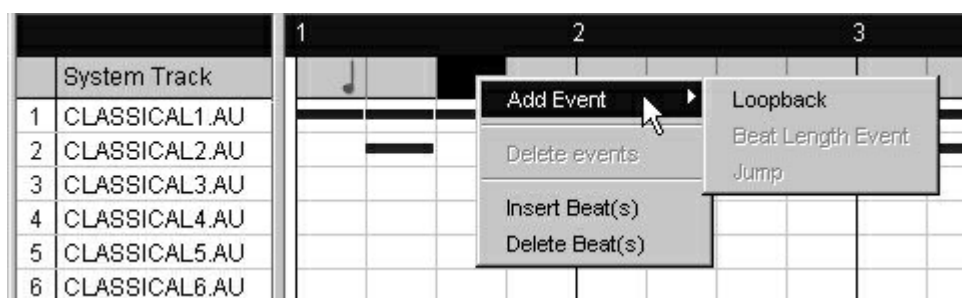
Double clicking in an empty beat will add the default event for that track. For normal audio tracks, that is a **Play** event; for the **System Track** it is a **Loopback point**.

### 3.6.1 The editing context menu

Clicking the right mouse button over the event list brings up the editing context menu. This will have one or more of the following options:

#### Add event

This leads to a submenu containing the events you can insert for this track:



In a normal track these will be **Play**, **Play once** and **Needed by**. In the system track this will be **Loopback**. Extra **Beat length events** and **Jump** are only available in the full version of Hyperceive. The appropriate event will be inserted at the beat you clicked on.

#### Delete event(s)

Delete all selected events

#### Insert beat(s)

Opens a small dialog (Fig 3.25) allowing you to specify the number of beats to insert, in all tracks, immediately before the beat you clicked on.

### Delete beat(s)

Deletes a selected beat or beats in all tracks.

### 3.6.2 Event-specific menu options

Some events will have specific options to manipulate them near the start of the editing context menu. These are described here.

#### Play and play once events

**Crop to here** sets the truncation point for this Play event to the end of the beat you clicked on.

**Shorten** - if this **Play** event currently runs for its maximum length, truncate it on the last beat. If not, shorten the playing by an entire beat. Truncation is indicated by a small red line at the point of truncation.

**Lengthen** - if this play event currently runs for less than its maximum length, moves the truncation point one beat further on. If this extends beyond its maximum length, sets it to play for its maximum length.

If one or more of these options is not available (for instance, lengthen would not be available if the play event already runs for its maximum length) then it will be present but not selectable.

### 3.6.3 Event properties

Most events have properties associated with them, and if this is so for the single selected event, the bottom option on the editing context menu will be **Properties**. Clicking on this brings up a small box allowing you to alter the properties of that event. Each event type that has properties is listed below, with a brief description of each property.

#### Play and play once events

**Beat** – the beat where this event has been set (uneditable).

**Beat length** – how long to play for, in beats.

**Play Once Only** – if set, this is a **Play once** event; otherwise it is a normal **Play** event.

#### Beat length events

**Beat** - the beat number at which the event is set (uneditable - in **H6** only one **Beat length event** is allowed. In the full version of Hyperceive, you can set as many as you like).

**Beat length** - the length, in milliseconds, at which to set the beat.

## **3.7 Testing**

You can test play your sequence before publishing it on the Internet to make sure that you are happy with the final result. The Play menu gives access to the testing features, and corresponds to the player functions on the tool bar.

### **3.7.1 The Play menu**

**Play** starts the sequence playing from the beginning. All clips are assumed to be loaded by the time they are needed, unless they are **Set unloaded** in the track list menu. The event list scrolls to keep pace with the playback. You can also start the sequence playing with **Ctrl Q**, or by pressing the **spacebar**.

**Play from selection** will start the song playing from the selected position.

**Pause** suspends the song in mid playback. To continue playing use the **Play** option.

**Rewind** will move the playback point to the first beat of the previous 4 beat bar (while the song is playing).

**Fast forward** will advance the playback point to the first beat of the next 4 beat bar (while the song is playing).

**Stop** stops playing, and returns the event list to the beginning of the song. Ctrl W or the spacebar are the keyboard shortcuts.

**Freeze** stops playing, maintaining the position in the event list. This enables you to stop anywhere in the song, without going back to the beginning.

**Scroll to start** scrolls back to the beginning of the song.

**Scroll to end** scrolls to the end of the song.

### **3.7.2 Modem simulation**

This function is only available in the full version of Hyperceive. It allows you to test your sequence under a variety of simulated Internet connection speeds. This way you can test your download times, **Loopback points** and **Needed by** events to ensure smooth playback in all conditions.

## 3.8 Publishing

When the time comes to publish your song within a web site, **H6** can help you ensure all the files you need are in place. In addition, it can optimise the **H6** song file for transmission over the Internet. The **Publish** option on the **File** menu does this. It opens a **Save as** dialog for you to give the file name of the published song file; it's a good idea to publish the song with a different name to that which you save it as normally, even if you publish to the same directory (see also section 3.9.5).

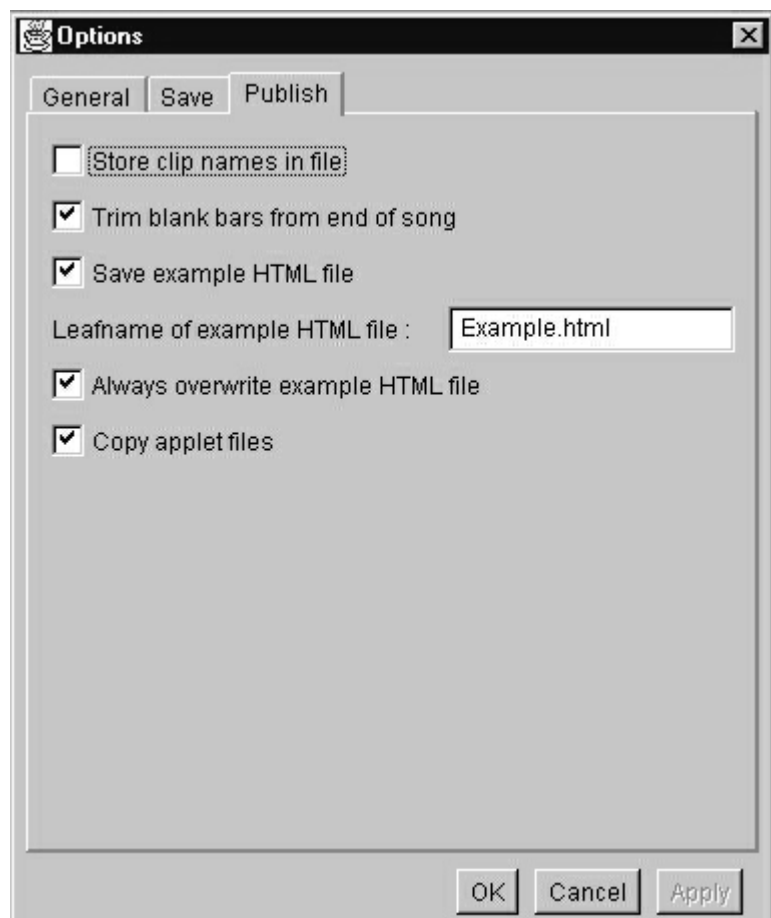
The editor will then publish your song by saving the file using the following options, which are available from the **Publish** tab of the **Preferences** dialog in the **File** menu (*right*).

**Store clip names in file** will store the names you have given to the tracks in the file. These names will take up space and are unnecessary for Internet transmission, so it's usually best to leave this box unchecked.

**Trim blank bars from the end of the song** will remove any unused beats at the end of your song, thereby minimising the size of the file.

**Save example HTML file** creates a sample HTML file with the correct markup to ensure that your song plays when the page loads. This can be used as a framework basis for the page you want the song to be contained in. The file is created in the same directory as the published song file. If this option is turned on, the name of the HTML file will be that given in the **Leafname of example HTML file** box. If the HTML file is already there, the editor will not create a new one, unless **Always overwrite example HTML file** is turned on.

**Copy applet files** copies the Java™ class files, which are essential to make the song play, into the same directory as the HTML file.





## 3.9 Preferences

In addition to options directly related to editing, **H6** has a number of options and features designed to make using it easier. These are described in this section.

### 3.9.1 General preferences

**Bubblehelp** is available on all toolbar objects, and anything that has extended information that isn't currently displayed, such as **Beat length events**, whose bubblehelp gives the beats per minute (bpm) of the beat length. To activate bubblehelp, leave the mouse pointer over the object for a second

Use **'bar n, beat m'** alternates between the two different ways of numbering the beats in a song. If this box is checked, the song will be numbered by four-beat bars; if the box is unchecked, the song will be numbered by individual beats.

**Audio clip editor** should contain the location of the audio file editor that you wish to use to edit your files. The editor is launched automatically when you double-click a file in the track list.

**Warn about invalid sample rates** is self-evident. If the box is checked, a warning will appear if you are using audio files that have not been sampled at the correct rate for use with **H6**.

**Autoscroll during playback** enables the player to scroll along the song as it plays, so that you can see what is happening in your sequence as you listen.

**Play startup sound** enables or disables the sound file which plays when **H6** is started.

**Audio directory** is permanently set to *Audio* in **H6**. In Hyperceive, it allows you to set the name that will be given to the directory in which the sound files you use will be stored on saving. This is also the name of the directory in which Hyperceive would look for the audio files when a song is opened.

**Configure modem simulation** only works in the full version of Hyperceive. There, it allows you to set the speed of the modem you wish to simulate when testing the download time of your song. This feature is not available in **H6**.

### 3.9.2 Save preferences

**Maximum recent songs on File menu** sets the number of songs that appear at the bottom of the file menu for easy reloading.

**Ensure audio clips accessible on save** makes **H6** check that your audio clips are all available for when you reload the file.



**Autosave every** sets the amount of time **H6** will wait before it automatically saves a backup of your sequence.

**Prompt on error** determines whether or not you are warned if there is a problem while autosave is taking place.

**Query revert to saved** will cause a prompt to appear when you choose **File → Revert**, making sure that this is in fact what you want to do.

### 3.9.3 Publish preferences

See section 3.8.

## 3.10 File layout

When you publish your **.hyp** song on the Internet, there are four types of file which must be present, these being the **.hyp** song file, the audio files, an HTML file and the **H6** player Java™ class files.

Due to the way the player works, there are certain restrictions on how you can lay out your files. The exact restrictions are given later but, for the moment, we'll consider only the suggested layout. This description assumes that you will be editing all your songs on one computer, and then transferring the files to a server to include them as part of your web site. If this is not the case, you will probably want to store the song files in the same directory as the HTML file and the Java™ class files.

Say we have a song file *Demo.hyp*, which uses two audio files: *Audio1.au* and *Audio2.au*. The suggested file layout is to have a directory *Audio*, in the same directory as the *Demo.hyp* file, and in which the audio files are stored. Let us say that *Demo.hyp* is in the directory *C:\mysite\Demo\*. Then the files would be in the following places:

```
C:\mysite\Demo\Demo.hyp
C:\mysite\Demo\Audio\Audio1.au
C:\mysite\Demo\Audio\Audio2.au
```

When you add a new audio file to a song, you can obviously add it from anywhere. When you save the song, **H6** will work out whether it will be able to find the file again; if it won't be able to, it will ask you what you'd like to do. The choices are to copy the file into the audio directory, to reference the file, in which case the editor will save the song as if it had copied the file, but won't actually copy it.

If you load a song which needs audio files which the editor cannot find, it will warn you, and draw the relevant track names in grey rather than blue. You will

then need to locate the correct file again, using **Track properties**. Note that unless you move or copy the audio file so that the editor can find it, you will need to do this every time you load the song for editing.

### Layout restrictions

The layout suggested above is designed to make it easy to work with **H6**, both in terms of editing and in subsequently publishing the song on a web site. It is mainly driven by the way Java™ restricts file access, which means that the **.hyp** player can only read files stored at or below the level of the Java™ class files. For example, if the main class file (*H6.class*) is accessed via the following Universal Resource Locator (URL):

```
http://www.mydomain.com/mysite/H6.class
```

then it can only load files whose URLs start with

```
http://www.mydomain.com/mysite/
```

So it could play the song

```
http://www.mydomain.com/mysite/mysong.hyp
```

using audio files stored in

```
http://www.mydomain.com/mysite/audio/
```

however it couldn't play the song

```
http://www.mydomain.com/mysong.hyp
```

(because it couldn't read the song file), and it couldn't play

```
http://www.mydomain.com/mysite/songs/mysong.hyp
```

using audio files stored in

```
http://www.mydomain.com/mysite/audio/
```

(because it couldn't read the audio files).

In addition to this, you should remember that the audio directory, and all file names, that you use will have to be given in a valid URL when you publish them on the Internet. H6 doesn't check that you're complying with the rules, since it would make editing more confusing, so you need to be aware of the issues. Essentially, the following characters are always 'safe' in a URL:

Alphanumerics (A–Z, a–z, 0–9)

Punctuation (-\_)

In addition, in the URL schemes you are likely to be using (http and ftp), ‘/’ is used to separate directories.

For more information, see RFC 1738 (*Uniform Resource Locators*. Berners-Lee, Masinter and McCahill).

## 3.11 Applet parameter reference

The **H6** player applet is controllable via parameters in the `APPLET` element of the HTML file which references the player:

```
<APPLET CODE="H6.class" ARCHIVE="player.jar" width=2 height=2>
  <PARAM NAME="audiobase" VALUE="Audio">
  <PARAM NAME="songfile" VALUE="songname.hyp">
  <PARAM NAME="colour" VALUE="0,0,0">
</APPLET>
```

### 3.11.1 Audiobase

The ‘value’ attribute of the audiobase parameter is the name of the audio directory, and therefore specifies the URL where the applet will look for audio files. In the case of Fig 3.36, the directory is called Audio. This may be an absolute URL; if not, it is taken relative to the document base. If that fails to provide a valid URL, the applet tries the directory audio/ at the document base.

Note that neither of the fallbacks is intended to provide an escape route for the situation where the HTML file contains an invalid parameter; they are intended to cover the situation where the document base passed to the applet is invalid for some reason.

### 3.11.2 Songfile

The ‘value’ attribute of the songfile parameter specifies the URL of the H6 (.hyp) song file to be used. This may be an absolute URL; if not, it is taken relative to the document base. If that fails to provide a valid URL, the applet tries the file song.hyp at the document base.

### 3.11.3 Colour

The colour is set as an RGB comma-separated triple (such as 255,255,255 for white). Typically this will be set to make the applet invisible within the page, that is, it will have the same colour as the background upon which it is set.

## **4 Tutorial**

In this section we'll go through an example project so that you can get an overall idea of the processes involved in creating a typical **H6**-enhanced site.

We'll use some simple music, and mix some voice announcements over the top.

## 4.1 Preparing your sound files

### 4.1.1 Recording sounds

Before getting to the **H6** stage, you'll have to record your music and voice sounds onto the hard drive of your computer. Wherever your sound files come from originally, you'll have to have them saved on your hard drive so that you can start work on them.

All multimedia computers have basic sound recording packages. Windows 95 has **Sound Recorder**. These can be used to record and save your files to your hard drive. However, as a user of **H6**, you will need to buy or download a slightly more sophisticated sound editing application for downsampling your sound files and saving them as Next/Sun **.au** files. Throughout this project, Cool Edit 96 is used as the sound editing application. It is shareware and is available on your **H6** installation CD. Please remember to register your copy of Cool Edit with the Syntrillium Software Corporation; you will then have access to all its features.

The CD player on your computer can be set up to allow the direct transfer of music to your hard drive. Also, your computer will have a slot into which you can plug your microphone, ready for recording. This input can also be used for recording from other media (e.g. tape, radio, etc.).

**IMPORTANT! REMEMBER THAT IT IS ILLEGAL TO REPRODUCE ANY COPYRIGHT MATERIAL WITHOUT THE COPYRIGHT OWNER'S PERMISSION.**

### 4.1.2 Saving sounds

The files used in this tutorial have already been prepared and can be found on your **H6** installation CD. The files for each stage have been saved, so you can practice each step and compare your results with those on the CD.

Create a directory for the purposes of this project, in this case (*see over*) we've called it *Voiceover*. It makes life much easier if all your files stay together. Now record your sounds and save them as **.wav** files into the *Voiceover* directory. **.wav** files are the Windows default and are easy to work with. In this case we've recorded our files as *voicemain.wav* and *synth.wav* and *piano.wav*.

When you record your audio files you will probably be asked at what frequency you wish to save them, whether you wish to record a stereo file and if you want to use 16 bit technology.

Computer audio quality is determined by the frequency at which the sound is recorded and played back. Modern CDs are mastered at 44.1kHz using 16 bit technology and are stereo files. At this rate, 1 minute of sound would take up 10 megabytes of computer memory. This is an enormous amount of memory, and is clearly not suitable for use on the Web. Nevertheless, at this rate the audio is clear, crisp and easy to edit.



In the context of this manual, editing your audio means preparing it for use with **H6**. **H6** uses files that are of a mono, 8 kHz, 8 bit format. So, through the editing process, your 44.1 kHz files are not only chopped to the desired length, but dropped to the correct sample rate, suitable for use with **H6** and the Internet.

### 4.1.3 Editing sounds

If we look at the music and voice files we wish to use, we can quite easily see how they naturally break into sections. The music can be separated into individual files: the synth and piano parts. Likewise, the voice can be separated into sentences with natural pauses between them.

The voice section reads:

*Hello and welcome to the site. Since you were last here we've updated the highlighted sections. Get in touch for details. Thank you for visiting.*

Notice that this breaks down into 4 sentences, which can be divided up into 4 separate files and then saved off for use in an sequence.

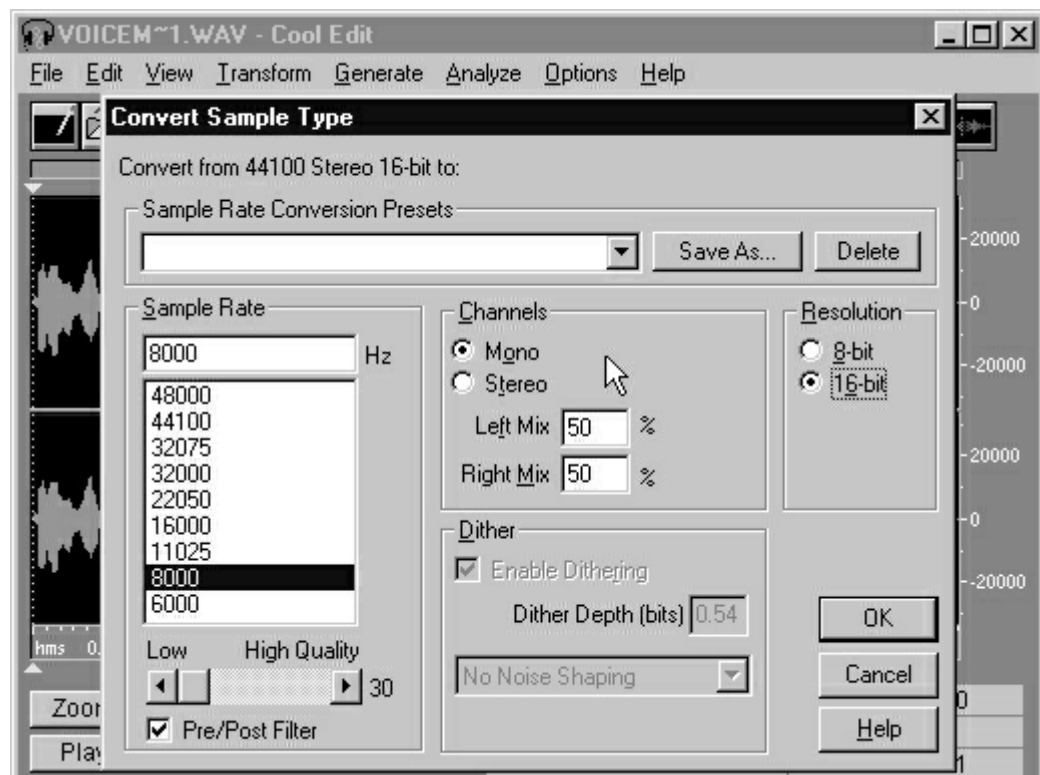
The section of music that we'll use consists of a background synth loop and a piano chord fill. This naturally gives two music files to be saved.

Using Cool Edit, or another sound editing application, you can now select and save these sections as individual files. The voice has been recorded as a single long file and saved prior to editing as *voicemain.wav* and can be found as *voicemain.wav* in the *wav stereo* folder of the *Tutorial* directory. If you load this into your sound editing application, you can listen to it being played back and at the same time get a visual representation of where the breaks occur (*see over*).

## 4.1 Preparing your sound files



This file is 16kHz stereo, with both left and right channels showing in the window. You must now **downsample** it to 8kHz 16 bit mono format. This is a simple process supported by Cool Edit and most other audio editing software via the **Convert sample type** function found in the **Edit** menu:





## 4.1 Preparing your sound files

You will now have an 8kHz mono file, with only one channel showing in the window:



The trick now is to highlight the desired section (*below left*), e.g. *Hello and welcome to the site* and copy it into a new working window (*below right*). Copying the selected section is preferable to cutting and pasting because the computer doesn't have to recalculate the original file after cutting. This can be a very lengthy and tiresome process, so copy your selection and paste it into a new file – it's much quicker. Apart from the speed factor, if you make a mistake with your editing you will always have the original to go back to.

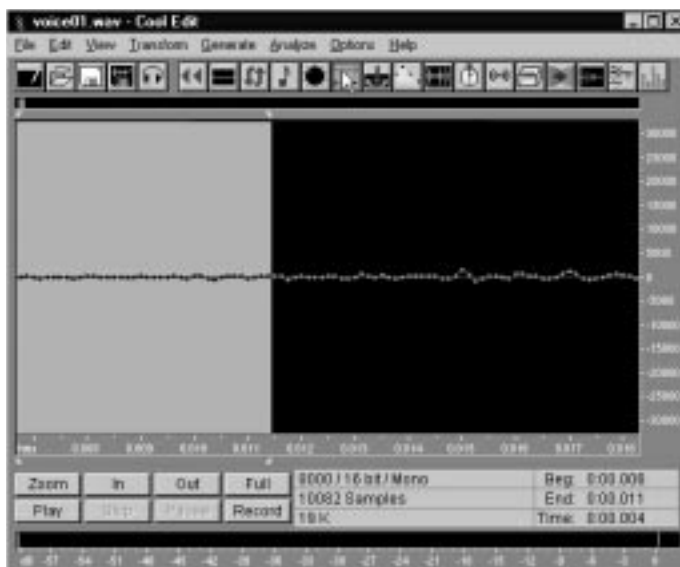
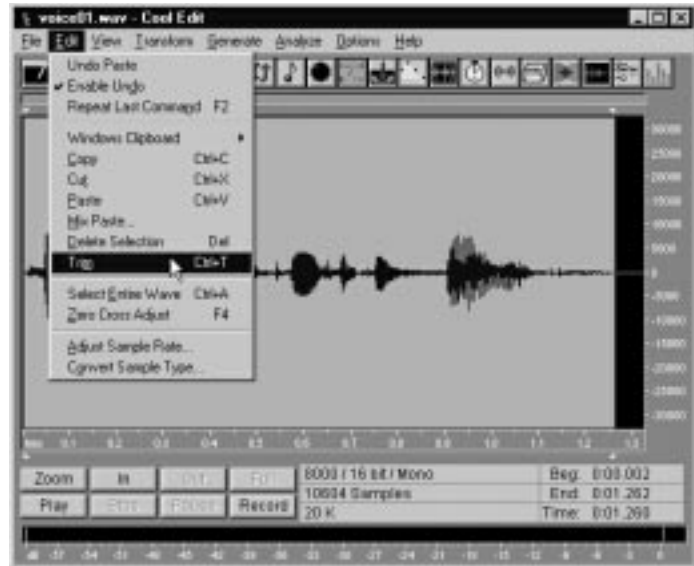




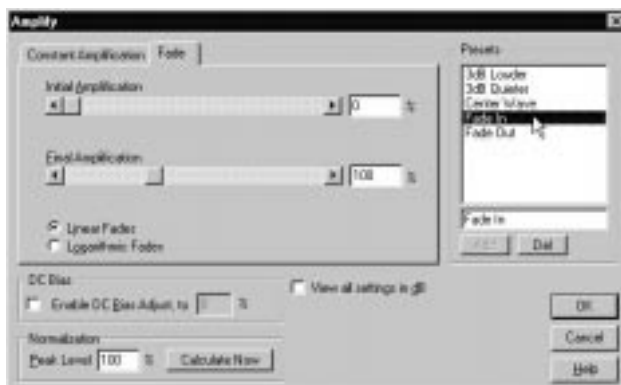
## 4.1 Preparing your sound files

In Cool Edit, you would select the desired section, copy it (**Edit** → **Copy**), then open a new window (**File** → **New instance**). Then paste the selection into the new window (**Edit** → **Paste**).

Once you have pasted your selection into the new working window, save it as *voice01.wav*. You will now need to top and tail the file. This is a simple operation that helps to make your new file sound clean. Most audio editing applications have a **trim** function within the **Edit** menu. Once you have highlighted the correct section of your file – only that part which contains the desired sound – hit the **trim** command and the outer edges, the unwanted bits, will be discarded (*right*).



In trimming your file, you may be left with a slightly jumpy start or finish. This is easily overcome by highlighting a tiny section at the very start of your file and performing a **fade in**. The figure below left shows a zoomed-in view of the very beginning of *voice01.wav* with the first part highlighted. The **fade in** function is a common command and will be located within the pull down menus of your audio editing application.



In Cool Edit, it can be found under **Transform** → **Amplitude** → **Amplify**. Once the start of your file is smooth, apply the same procedure to the very end of the file, this time using the **fade out** function.

From here it only remains for you to save your file as a Next/Sun .au file, which automatically produces an 8 bit version, and it will be ready for use with **H6**. Make sure that **mu-law 8-bit** is selected in the **Options** tab in the **Save** dialog when you save your file. Other types of .au file will not work with **H6**.

Once you have successfully completed this process with one file, you'll find it becomes a lot easier. You'll now be able to repeat the process with the other files that you require for this project.

In the directory you made for this project, you should now have your individual audio files. Here we have: *01synth.au*, *02piano.au*, *03voice.au*, *04voice.au*, *05voice.au* and *06voice.au*.

Now it's time to launch **H6** and start sorting your files into a working sequence.

## 4.2 Arranging your sound files

### 4.2.1 Inserting files

For details on how to insert the audio files, please refer to section 2.3.

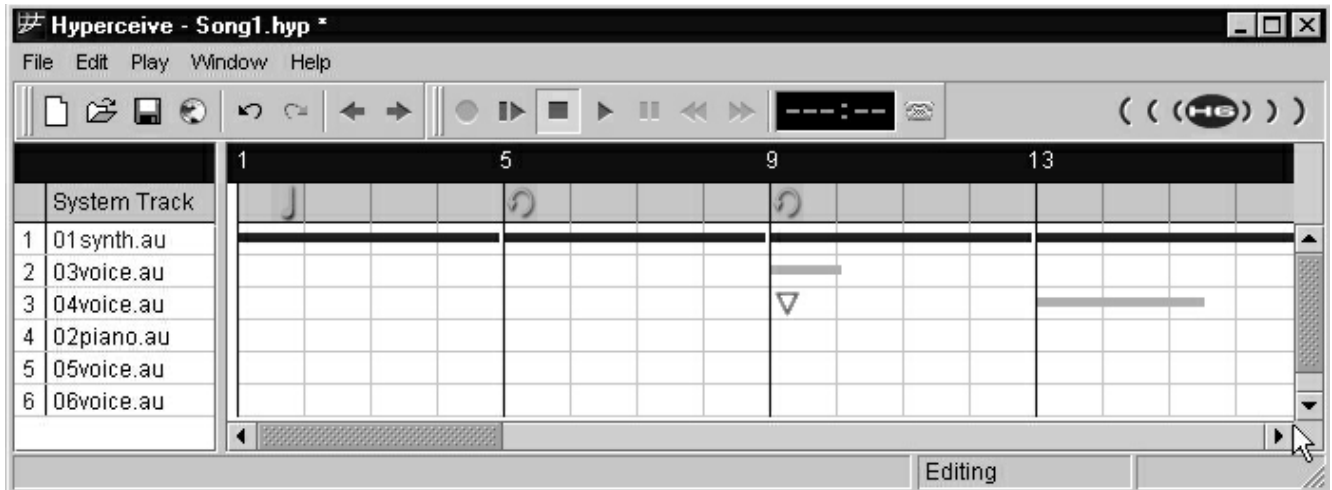
For this sequence, we want the synth to start playing, then 2 of the voice files to come in over the top of the music, followed by the piano and final 2 voice files. To allow this to happen, load the audio files (see section 2.3) in the following order :

- |          |                   |          |                   |
|----------|-------------------|----------|-------------------|
| <b>1</b> | <i>01synth.au</i> | <b>4</b> | <i>02piano.au</i> |
| <b>2</b> | <i>03voice.au</i> | <b>5</b> | <i>05voice.au</i> |
| <b>3</b> | <i>04voice.au</i> | <b>6</b> | <i>06voice.au</i> |

**H6** will select a beat length value which, by default, will be the length in milliseconds of the first file (*01synth.au*). This can be changed if necessary (see section 2.5).

Once the files are loaded, you can insert the beat positions of each track, appropriate to your sequence.

In the figure overleaf, *01synth.au* has been inserted repeatedly throughout the sequence as a loop. You may have to adjust the beat length to make the loop run smoothly (see section 2.5). For this loop to work and to set the timing so that the other files could be placed in the right places relative to each other, it was necessary to reduce the beat length at beat 1 to 1090ms (using the preprepared *01synth.au* file available in the *Tutorial* folder in the **H6** program directory). For *01synth.au*, this produces a smooth loop.



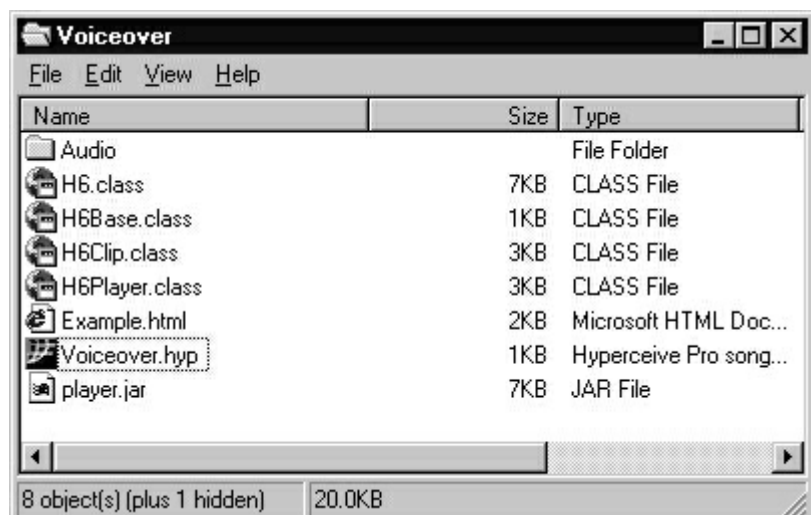
The voice files have been inserted as **Play once** events (see sections 2.6 and 3.5.2). **Needed by** events have been entered to ensure that certain files have loaded and are ready to play by the time those **Needed by** signs are reached (see sections 2.6 and 3.5.2). In this case, we've added a **Needed by** event for illustration. In reality it isn't essential that *voice2.au* has loaded by the time *voice1.au* begins to play - the pause between the two sentences isn't critical. **Loopback points** have been added to tell the player where to jump back to if a file hasn't loaded in time due to a slow Internet connection (see sections 2.5 and 3.5.1).

## 4.3 Publishing

When you are happy that your sequence works, it's time to publish it and incorporate it into the Web page of your choice. Use the **Publish** function to save all the necessary files to your project directory and to prompt **H6** to generate the example HTML file (see sections 2.8 and 3.8).

The figure shows the contents of the project directory with all the files that the publish function has placed there. These are the *Audio* directory with all of the sound files used in the sequence, the four Java™ class files and the *.jar* archive, the example HTML file (here called

*Example.html* - but can be changed in **File** → **Preferences** → **Publish**), and the *.hyp* file (*Voiceover.hyp*).



The *player.jar* file is a compressed archive of the Java™ class files. The latest Web browsers recognise the *.jar* file format, and will download that in preference to the four individual class files. This reduces the download time because the *.jar* file is smaller than the four class files combined. The individual class files are still included for the sake of older browsers.

Shown below is the part of the example HTML file (*Example.html*) that is needed in the Web page to activate the sound. This is this part which you should copy and paste into your site.

```
<!-- Hyperceive playback system (c) Copyright Insignia Technologies Ltd. 1999 -->
<APPLET CODE="H6.class" ARCHIVE="player.jar" WIDTH=2 HEIGHT=2>
  <PARAM NAME="audiobase" VALUE="Audio">
  <PARAM NAME="songfile" VALUE="Voiceover.hyp">
  <PARAM NAME="colour" VALUE="0,0,0">
</APPLET>
```

You can view *Example.html* in your browser before you paste the applet information into your page, if you want to check the sound first.

## 4.4 Orchestrating your site

The sequence that we have looked at in this chapter could quite easily sit behind a Web site such that the audio content continued regardless of how the site was navigated. To do this you would use the persisting frame idea. This involves building a blank HTML page and including it in your frameset, sizing it

```
<HTML>
<HEAD>
<TITLE>My H6 Site</TITLE>
</HEAD>

<FRAMESET COLS="0,177,*" FRAMEBORDER="no" BORDER="0" FRAMESPACING="0">
  <FRAME SRC="H6.html" SCROLLING="no" RESIZE="no">
  <FRAME SRC="navbar.html" SCROLLING="no" RESIZE="no"
  <FRAME SRC="home.html" SCROLLING="auto" RESIZE="no"
</FRAMESET>
</HTML>
```

to '0' so that it remains invisible:

Alternatively, you might want to use this audio somewhere in the site such that it is activated by arriving at the appropriate page. This involves either cutting and pasting the applet tags and **H6** information into the HTML of that page, or by building onto the example HTML page produced by the publish function of **H6**.

Hopefully, you've kept all your files in the working directory for this project. You will see that, providing you have saved your work correctly, your **.hyp** file (the sequencing information) is also in this directory, as are the 5 applet class files. All you have to do now is FTP your complete page up to your server space, together with the Audio directory containing the audio files, the **.hyp** file and the 5 applet class files.

# Glossary

## Glossary

*For explanations of specific H6 functions and features, please refer to the main part of the manual. Words in bold refer to cross references within the Glossary.*

### Applet

An applet is a small program designed to run from within another program. On the World Wide Web, there are many applets written in **Java™**, which can download with a Web page and run on the **end user's** machine. Applets can be written to perform a multitude of functions to enhance a site. **H6** is powered using applets.

### Sequence

See sequence.

### Audio directory

The directory in which **H6** stores the audio files used in a particular **.hyp** file when that file is saved or published. It can be given any name and is configurable in the Preferences dialog.

### Audio editing package/software

An application such as Cool Edit with which the user can edit audio files. The user can also add effects to the audio.

### AU files

This is the standard audio file format for the **Java™** programming language. AU files usually have the extension **.au**. **H6** works using **Java™**, and therefore it uses **.au** files

### Backing track

A sound sequence that runs constantly in the background, regardless of what the user is doing. It is usually used to add atmosphere to a site.

### Bandwidth

The range of transmission frequencies a network can use, expressed as the difference between the highest and lowest frequencies of a transmission channel. The higher the bandwidth, the more information can be transmitted at once, or the faster the transmission. The more complex the data, the more bandwidth is required. For example sound files and image files require much more bandwidth than simple text files. Normally continuous audio on a site

would require a lot of bandwidth to be possible. This is not the case with a **H6**-enhanced site. **H6** minimises the bandwidth requirement and doesn't clog up the net in the same way as streaming technologies.

### Bar

In **H6**, a division of four **beats**.

### Beat

A single division of a **sequence** having a specific unit time value. In **H6** a beat is the area between two vertical lines in the main sequencing window, and the length of the beat is determined by the value set in the previous Beat Length Event.

### Bit

A single unit of digital information expressed as either 1 or 0. The bit depth of a **sample** affects the quality of the sound – 16 bit samples being twice the quality of 8 bit samples.

### Cache memory

A section of a computer's memory in which frequently-used information is stored for fast access. When someone looks at a **H6**-enhanced site, each AU sound file is stored, once it has **downloaded**, in the cache configured through the end user's browser. The files can then be played from the cache, which is much quicker than downloading the files each time.

### Context menus

Pull down menus which present options to the user which are dependent on the circumstances in which they were displayed. Context menus are usually accessed by clicking the right mouse button.

### Document base

The directory in which the document is contained.

### Downloading

To copy data (usually an entire file) from a main source to a peripheral device. The term is often used to describe the process of copying a file from an online service to one's own computer.

### Downsample

Reducing the sample rate and/or bit depth of a sample.



### Editor (H6)

Used in this guide to describe the functions of **H6** which are involved with editing a sequence, as opposed to the **Player** functions of **H6**, which deal with playback.

### End user

The person who uses the final version of a product or service. With regard to the **Internet**, it refers to anybody who surfs the Web and visits a site.

### Event

In **H6**, a specific function which has been set by the user to have an effect on a **sequence**. Track events include Play, Play once and Needed by. They determine when and how a file will play in the sequence. System events include Loopback points and the Beat Length Event. They control the way the player plays back the sequence.

### Event list

The part of the editor window which displays where the events have been entered (see section 3.5).

### Fade in/out

Functions performed by an audio editing package in which an audio file is made to start and end smoothly. A tiny section at the beginning of the file is selected and the Fade in function applied. This ensures that the file begins at a volume level of zero, eliminating any jumpiness. A similar process smooths out the end of the file using the Fade out function.

### Frames

Most modern Web browsers enable Web page designers to divide the display into two or more sections, called frames. Each frame contains a different page. Often one frame is a navigation bar or menu bar which persists throughout the site. The user selects pages to view from this bar, and the selected pages appear in another frame.

### Frequency

See sample.

### HTML

HyperText Markup Language. The language used to create Web pages. It includes instructions for formatting text, creating links to other pages or sites, inserting images and applets etc.

### **.hyp**

**H6** files are saved in a unique Hyperceive format. They have the extension **.hyp**.

### **Internet**

A global network connecting millions of computers. As of 1998, the Internet has more than 100 million users worldwide, and that number is growing rapidly.

### **Java™**

A programming language for Internet and intranet applications developed by Sun Microsystems. Java™ has a number of features that make it well suited for use on the Internet. Small Java™ applications are called Java™ **applets** and can be downloaded from a Web server and run on your computer by a Java™ -compatible Web browser, such as Netscape Navigator or Microsoft Internet Explorer. H6 itself is written in Java™, and the elements that it produces to allow sound on a Web page are also in Java™.

### **Java™-enabled browser**

A Web browser which is able to run **Java™ applets**. All versions of Netscape Navigator since version 3.0, and all versions of Internet Explorer since version 3.1 are Java™-enabled.

### **Loop**

An audio file that, when repeated continuously, will give the impression of being a single continuous file. A typical example of a loop is a drum loop in which a small sample of a drum pattern is repeated given the impression of a constant rhythm.

### **Loopback**

If an audio file has not **downloaded** by the time it is required in the sequence, H6 will loop back to a predetermined point so that there is no interruption to the sound. When the file has downloaded, the sequence will continue.

### **Next/Sun .au files**

See AU files.

### **Off-beat**

A measure of time falling exactly between two down beats of a bar.

### **Offline**

Not connected to the **Internet**.

## Online

Connected to the **Internet**.

## Persisting frame

A **frame** which remains open permanently. Sound can be made to continue through a framed Web site regardless of the visitor's actions by placing the **H6** sound information in a persisting frame. This is sized to zero, so is invisible to the visitor, and cannot be closed.

## Player (H6)

Used in this guide to describe those functions in **H6** which control the playback of a sequence, as opposed to the **Editor** functions.

## Plug-ins

A piece of software that extends the functionality of an application. **End users** usually have to download plug-ins to enable their Web browsers to display certain types of file, especially multimedia files. **Java™-enabled Web browsers** do not need any plug-ins to play the sound on H6-enhanced sites, so the **end user** can enjoy it without needing to download any extra software first.

## Preload

To **download** a file in advance before it is used.

## Sample

1 *v.* A technique used to record continuous phenomena, such as sound. Many individual samples are taken, the **sample rate** of which determines the quality. Motion picture recording is an example of the principle of sampling, with many single frames being recorded at a high rate. When played back they give the illusion of continuous motion. 2 *n.* A sound file which has been recorded electronically in this way. Often this will be of a specific sound type, e.g. a drum sample.

## Sample rate

The frequency with which each individual recording is made in a sample. The higher the frequency, the better the quality of the sound. It can be likened to the frames per second value in motion picture recording. Sample rate is measured in Hertz.

## Sequence

A collection of instructions which determine how a group of files such as audio

files should play together to produce a complete tune, voiceover, etc. The **.hyp** file that H6 produces is a sequence file. It contains information such as when to play which file and at what speed to play it. All the functions available in the H6 editor can go towards determining the sequence.

### Shareware

Software which is distributed free on a trial basis. After the trial period, the producer requests that a small fee is paid if you intend to continue using the program. By paying the fee and registering as a user, you are usually entitled to upgrades and support.

### Song

A new **H6** document is referred to as a New Song. In this case, song is another word for **sequence**.

### Soundtrack

The total collection of sounds and music which accompany a Web site (or, for that matter, any multimedia presentation). It could be made up of one **sequence** or a number of them.

### Streaming

A technique for transferring data as a continuous stream from one place to another over a network. it is usually used for large multimedia files. Each time a file is sent in this way, it requires a specific section of **bandwidth** to be taken up until the transfer is complete. This means that the Internet gets clogged up if a large number of people are using streaming technology to transfer files. H6 does not require streaming technology to deliver continuous sound to a Web site. It uses any available bandwidth and does not clog up the Internet.

### Throughput

The speed at which files are downloading from the Web. It can vary enormously according to the speed of connection, the time of day and a host of other factors.

### Track

The horizontal space alongside an audio file in the H6 window in which **events** are entered which relate to that file.

### Track list

The list of files used in an **.hyp** file which appear in the left-hand section of the **H6** editing window.

## **Trim**

A function found in audio editing packages by which unwanted parts at either end of an audio file can be removed.

## **URL**

Uniform Resource Locator is simply the address of a Web site, such as [www.hyperceive.com](http://www.hyperceive.com).

## **WAV files**

The default audio file format in Windows.