SemanticHIFI

IST-507913

Public Report of WP6 "Authoring"

Covering period: December 2003 – November 2006

Report version: **1.0**

Report preparation date:

Writers: **Andre Graef, Florian Plenge, Norbert Schnell**

Control: **Hugues Vinet, IRCAM**

Classification: **Public**

Contract start date: **December, 1st 2003**

Duration: **36 months**

Project co-ordinator: **Hugues Vinet, IRCAM**

Involved Partners: Native Instruments (WP Coordinator), IRCAM, UPF

Project funded by the European Community under the "Information Society Technology" Programme

Table of contents

1 WP Overview

1.1 Objectives

The objective of work package 6 is to develop a real-time mixing and non-real time mix editing software for the advanced home user to the pro-DJ with easy-to-use intuitive user interface, the so called Authoring Application. This includes the development of functional modules in WP6 as well as the integration of modules developed in other WPs to provide or enhance browsing, searching, sharing, mixing and performing features.

1.2 Partners' roles

NI : WP leader, technical development and integration, development of specific functional modules (real time mixing, non-real time editing, loop processing), user tests, documentation

IRCAM : development of specific functional modules (real-time and non real time audio processing functions) and adaptations of Indexing (WP2) developments according to implementation constraints.

UPF, SONY-CSL : adaptations of Browsing and Performing developments according to implementation constraints.

FhG : adapted implementation of audio data and metadata management modules – coordination with Sharing server development.

1.3 WP contribution to the project

Development of a real-time mixing and non-real time mix-editing software. Features provided exclusively by the Authoring Application

- Application complementary to the Hifi System for the advanced user
- Advanced Interaction with musical content (one source).
- Mixing using advanced effects and multiple decks
- Content creation (playlists, non-destructive mixes, radio shows)
- Convenient administration of the music collection
- Making content and meta-data available for other users via p2p

1.4 Synthesis of main achievements

Final functional prototypes concerning the core application (WP6.1) were developed and documented. These prototypes provide functionalities to playback audio-files, to navigate in a song, to manipulate audio-material and to create mix-files in real-time and to edit Mixes afterwards in non-real time. Beyond that the Technical Framework necessary for the integration of the partner modules from other work packages was developed (WP6.2). In essence these modules are a "VST-Host" (in order to integrate real-time audio manipulating modules), "Sharing Component" (in order to establish a connection to the Sharing System), and a web host including socket API (in order to host multimedia content provided by the partners). The focus of the work was on user requirements, use cases, user interfaces, the software architecture and general framework, specification of interfaces for the integration of the modules of the other partners, and the development of specific functional modules. User feedback resulted in a re-specification of which central elements were implemented in the last phase of the project. Extensive documentation was provided during the project lifecycle from technical to administrative and public level.

2 WP Results and Achievements

2.1 WP6.1: Core Application

Responsible partner : Native Instruments

2.1.1 Functional description

The purpose of this module is the (further) development of a software framework and of functional modules for real-time and non-real-time audio processing with the purpose of musical content creation, potentially recorded as so called Native Mixes. In the following all features developed for advanced real-time mixing in the frame of Semantic Hifi are given. These features are integrated in one final application and have been developed completely new in the frame of SHF.

2.1.1.1 Advanced EQ and Master section

The advanced EQ and Master section is the central mechanism to control the audio-output of the Authoring Application. All audio-streams of the Decks used by the user and the external audio inputs, eg, mic inputs, can be routed into this section and manipulated by its extensive set of parameters. The EQ and Master section consists of a perfectly emulated Allen&Heath Xone:92™ 4-channel club mixer, including its adjustable cross-fader curve, cross-fader assignable filters and high-end EQs. Since the Authoring Application provides 4 Decks to playback tracks, a central control panel was developed which allows the user to assign all Decks in a free definable combination to the Crossfader. The EQ and Master section itself is also free customizable. Each of the 4 controls from the EQ-Control to the Master Control can be turned on and off, which enables the user to design his environment intuitively. Extending state-of-the-art applications the EQ section is exchangeable. The user can choose from 4 different types of EQs which are outstanding emulations of real existing DJ-EQs like the Ecler Nuo4™. Also different cross fader curves are available.

2.1.1.2 4 fully-featured playback decks including an optional 2 deck mode

With the Decks the Authoring Application provides 4 completely new designed virtual turntables that behave like any real DJ turntable. In order to playback audio-files the user can now comfortably load songs into a deck (e.g. by drag & drop). The graphical representation of the track is generated, and displayed in 2 different ways, which give more control of the audio file to the user as state-of-the-art applications do. The Waveform Display gives a detailed overview of the audio material in a range the user can define himself. In the Stripe Display the user gets an overview of the whole track. By the usage of this display the user can navigate intuitively in the track and set Cue Points and Loop Points (see below). All 4 Decks can be used simultaneously. Each of these Decks consists also of control-elements, which enable the user to synchronize the decks and to modify the playback (speed, loops) and the pitch (key) of a track. Decks are the central mechanism to playback audio-files in the Authoring Application.

2.1.1.3 Advanced remixing functionality (loop processing)

The Authoring Application provides new real-time loop editing features which extend state-of-the-art looping features. One way to loop in a track is the Loop Points feature. The user sets a point in a track that defines the start of a section to be looped by pushing a single button on the Deck. Since the loop-length is definable in relation to the beat-grid, the end-point is set automatically and the loop runs seamlessly. Once the Loop is set the entire loop interval can be moved, even while the track is playing. Beyond that the loop-length can be modified afterwards directly at the respective deck.

2.1.1.4 Composition of segments: real-time mixing

The Beat Masher is a unique effect that isn't based on any classic effect type. It essentially samples a bar of music into a buffer which can then be transformed, and mashed. The buttons and knobs of this effect have to be explained thoroughly and in detail:

• Tap sets the tempo when the Beat Masher is not synced to the deck or the Master. If the effect is running in Sync, this button has no effect.

• Action starts the sampling until the buffer is full. Then, it repeats the recorded audio and warps it accordingly to the settings of the effect.

• Length defines the length of the Loop recorded in the buffer. The amount is always based on beats, and from left to right the values are: 1/32th (minimum value), 1/16th, 2/16th, 3/16th, 1/8th (centre position), 3/8th, 2/4th, 3/4th and one bar (maximum value).

• Sync synchronizes the tempo to the deck or to the Master if the Beat Masher is used as a master effect.

• Rotate changes the position of the Loop within the sampled bar. This function is most effective at short to minimum setting.

• Reverse plays the Loop backwards. If this is combined with a Gate Value set between 8 am and 10 am the effect is very obvious because the original signal is being punctured by short bursts of the reversed Loop.

• Gate works in two different modes. If you move it from the centre towards

the maximum value, it progressively mutes sections of the Loop until only one 16th of the Loop is audible at 100%. When moved from the centre towards minimum value, the original signal is being mixed into the loop, resulting in the most interesting effects. If Gate is in the centre position, it plays the Loop exactly as defined by the Length knob.

Note: You will not be able to get the optimum result out of the Beat Masher effect without triggering it by hotkeys or with the help of a MIDI controller, as this effect develops its character only when several parameters are being tweaked at the same time

2.1.1.5 Seamless navigation in a track

The user can navigate on the beat-grid of a track calculated during the indexing step using the Snap Mode in the Stripe Display. Clicking a mouse-button the user jumps to a beat-marker at the respective position in the track while keeping the beat. With the Beatjump feature Native Instruments developed a new groundbreaking feature that allows the user to jump seamlessly through a track in multiples of beats. The user can specify how many beats forward or backward he wants to jump. This can be very useful for scrolling through a track, but also has an added effect of remixing when performed while the track is playing. The segmentation markers calculated during the indexing step (WP2) are stored in xml format in the collection NML and in the tracks as Cue points. The user can navigate the segments via the familiar cue point panels and controls.

2.1.1.6 Real-time processing (tempo-synced FX and filters)

For the manipulation of audio-material played on a Deck the extensive FX and Filters section provides a set of state-of-the-art effect and filter modules. Since these effects and filters can be synchronized with the tempo detected during the indexing steps the user can integrate these features comfortably in his live-performance or mix-recording. To increase the flexibility to control these effects or filters the user can modify its parameters directly at the mixer section or at separated control panel. The FX contributed by the partners (Groovator, Instumentizer and Xmix) are also controled via these controls. The advanced user can open additional user interfaces dedicated to these advanced FX.

2.1.1.7 Recording all mix parameters in a non-destructive mix-file

A performing session with the Authoring Application can be recorded as a so called Native Mix saving all DJ moves as control data. For the first time a DJ mix can be recorded as a non-destructive, editable control file. The original tracks remain untouched by the mix. It allows to save hours of performance in a file size suitable for convenient sharing and distribution over the internet. Additionally, it allows convenient dubbing of a mix and to manipulate the mix in non-real time using a graphical user interface.

During playback of the mix the individual tracks together with the control data is rendered to give the original mix. The actions performed by the artists during the recording can be displayed while the pre-recorded mix is played back. A mix can be shared on the sharing system and can be played back if the retrieving user possesses the identical tracks as the creator – without infringing intellectual property rights. Mix playback module

The pre-recoreded mixes can be loaded and played back with this module if the tracks used to create the mix are also present. The tracks are linked in the mix using a special Audio ID created solely for this special purpose. The mix playback module is running on the HIFI System and in the Authoring Application. OSC commands are used as API to control the mix playback.

2.1.1.8 Non-real time Mix-Editing Feature

This module provides functionalities to visualize and edit recorded mixes in non-real time extensively. Control values as well as time and position information and the tempo of the tracks can be edited, inserted or deleted in a intuitive way combining the feeling of a live DJ performance and familiar stuido production tools to a completely new way to interact with audio material.

2.1.1.9 Publication of Playlists

The publication of playlists as html files has been implemented. The user can customize the fields he would like to see in the playlists (track number, duration, artist, title, ...) and their order. XSLT is used to transform the XML playlists of NI to html code. This enables advanced users to customize the printed playlist in any form he wishes and facilitates the publication on the internet and via P2P networks.

2.1.1.10 Sharing

An intuitive user interface for the execution of the search request, presentation of search results and downloading of files has been designed and implemented. Additional features like the Search Result Preview let the user get information about the content of a Playlists or Mixfile in advance

2.1.2 Method description, position over state of the art and implementation

WP6 is an application workpackage. Industry standard software development practices are employed for the development of the Authoring Application. It is a multiplattform development (Win XP, Mac OS, Linux) in C++ and Java. Native Instruments REATKOR is used for prototyping the DSP modules. Detailed UML diagrams and APIs are given in D1.2.4. Methods beyond state of the art were implemented to realize the Native Mix recording and playback. See Annex for details. With the integration of a sharing-component into a DJ-application a completely new path was taken. There is no other state-of-the-art DJ-application on the market which implements a feature that allows the sharing of Playlists or Mixfiles. During the User Feedback Session it showed up that users are above all interested in the sharing of mixfiles.

For the development of the semantic audio processing module prototypes the Max/MSP environment on Mac OS X (www.cycling74.com/) was used to achieve rapid and flexible prototyping of the user interaction, including the involved signal processing functionalities and layout of the user interface. For the optimization of the prototype modules, core signal processing routines are taken from a library of modules developed for Max/MSP using the standard Mac OS X development tools.

2.1.3 Benchmarks

WP6 is an application workpackage. Technical benchmarks for the application and the functional modules that are beyond state of the art can not be given since they are unique. Benchmarks in a broader sense are the competitors in the market place. An overview is given in the Annex.

2.2 WP6.2: Technical Framework

Responsible partner : Native Instruments

2.2.1 Implementation

The purpose of this module is to create an interface to external services from other work packages, and integration of specific modules in the Authoring Application.

2.2.1.1 Connection to the Sharing System

The Connection to the Sharing System was established by the implementation of the Sharing Component, which enables the direct communication between the Authoring Application and the Sharing Client provided by the Fraunhofer IDMT. All functionalities for publishing, searching and downloading of all permitted file types (Playlists, Mixfiles, Musictrack metadata) were implemented..

2.2.1.2 Multimedia rendering - Embedded Web-Browser and Socket communication

A Web-Browser has been developed and embedded, which enables the Authoring Application to display web-content in a custom window. This component is used to integrate the Sharing Component. For the communication between the Authoring Application and the Sharing Component both applications contain socket-objects.

2.2.1.3 Protocol to integrate real-time audio manipulation modules from the partners

The new VST-Host provides the fundamental functionalities and mechanisms to integrate external modules which manipulate audio-material in real-time. These modules, so-called “VST-Plug-Ins” were provided by the WP-partners IRCAM and UPF. Just as the internal effects of the Authoring Application these modules can be selected comfortably from a list of available modules. To modify parameters of the VST-Plug Ins the user has access to the GUI of the respective Plug-In-GUI. Furthermore a mechanism to synchronize Authoring Application and VST-Plug Ins has been implemented utilizing the standard mechanisms provided in the VST API (http://www.steinberg.de/DocSupportDisplay_sbf02f.html?templ=&doclink=/webvideo/Steinberg/support/doc/VST%20Audio%20Plug-Ins%20SDK_1.htm&Langue_ID=&Product_ID). To integrate the “Instrumentizer”-Plug-In a microphone input has been implemented.

2.2.1.4 OSC – Open Sound Control

Using the network protocol OSC, controllers, multimedia devices and applications can comfortably be connected to the Authoring Application. This protocol operates at broadband network speed and allows new types of realtime interactions which are not possible with the usual MIDI standard. With OSC the user has more flexibility in the kinds of data he can send.

2.2.2 Method description

Sharing component

The Sharing Component enables the direct communication between the Authoring Application and the Sharing Client provided by the Fraunhofer IDMT, and consequently with the entire network. It acts as a second middleware which can process and forward all queries and messages from and to the Sharing Client. The communication between Authoring Application and Sharing Client is handled by sockets. The mechanism for obtaining the TUIDs was implemented by the establishment of a connection with the AudioID-Server as well as the integration of the “xtrmain.exe”-module, which generates the necessary fingerprint-files.

VST-host

The VST-Host was implemented from scratch. A FX or Instruments Plug-In is loaded each time the user selects it from the dropdown-list available in the new implemented Channel-VST module. After activating the Plug-In by pushing the ON button the audio-buffers of the associated Deck and the Plug-In are transferred. To determine the current position in a track, which is indispensable for the integration of IRCAMS Plug-In “X-Micks”, the tempo information calculated in the earlier indexing steps is passed to the plugin via the VST API. In order to integrate the “Instrumentizer”-Plug-In delivered by the UPF a microphone input was developed. This signal is routed to a Deck which sends the signal to the Plug-In.

2.2.3 Position over state-of-the-art

WP6 is an application workpackage. Industry standard software development practices are employed for the development of the Authoring Application. It is a multiplattform development (Win XP, Mac OS, Linux) in C++ and Java. Detailed UML diagrams and APIs are given in D1.2.4.

2.2.4 Benchmarks

In the context of the SemanticHIFI-project the Authoring Application with its new implemented features is a prototype which is difficult to compare. Since there are no other applications which are similar with the Authoring Application that implement these new features, a comparison was not feasible.

2.3 X-Micks Song Morphing Plugin

Responsible partner : IRCAM-ATR

2.3.1 Functional description

The *X-Micks Song Morphing Module* allows for interpolating and mixing of two songs in the authoring application. Using the module the user can create transitions from one song to another and create hybrids of two songs other than by usual cross-fade.

The module has two stereo inputs for to two beat-synchronised song inputs and provides one stereo output with the hybridised sound. Even if the control of module is slightly more complex than a cross-fader, a simple and intuitive user interface (see figure 1) has been developed for the module.

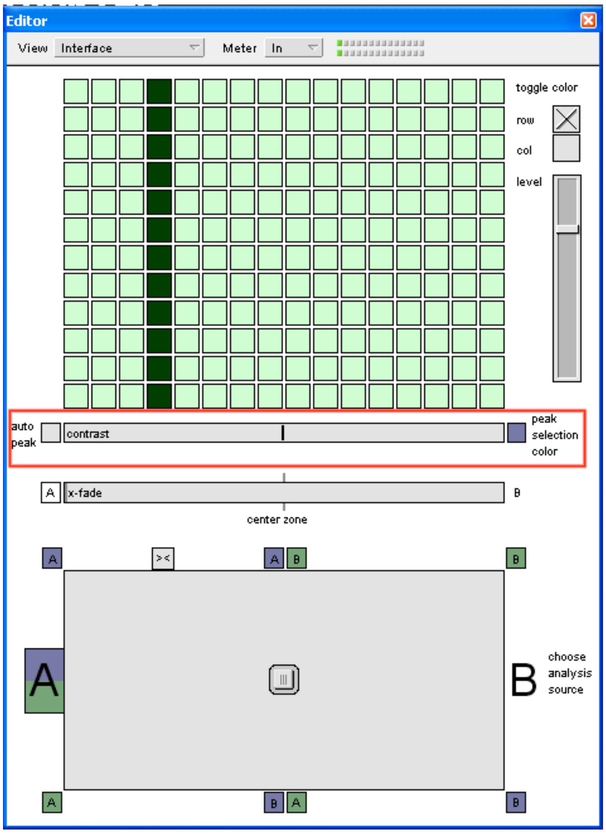

Figure 1: X-Micks VST plugin interface. Highlighted are the a*uto peak* function and upper fader

The chosen approach is based on beat-synchronous band filtering: The *X-Micks* module decomposes the incoming sounds into 12 perceptive filter bands, which are controlled beat by beat synchronously to the incoming soundfiles. The user interacts with the module via a matrix display representing a beat pattern (typically 16 beats).

Each square of the matrix represents a certain frequency band at a certain beat in a rhythmic pattern of the beat-synchronised audio streams input to the module. Selecting and deselecting the squares of the matrix, the user can chose for each beat, in which way the audio output will be composed mixing frequency bands from the audio inputs. The two audio inputs and the filters controlled by the matrix are beat-synchronized.

The actual mixture of the two input sounds represented by the selected and unselected matrix squares is determined by a second interface.

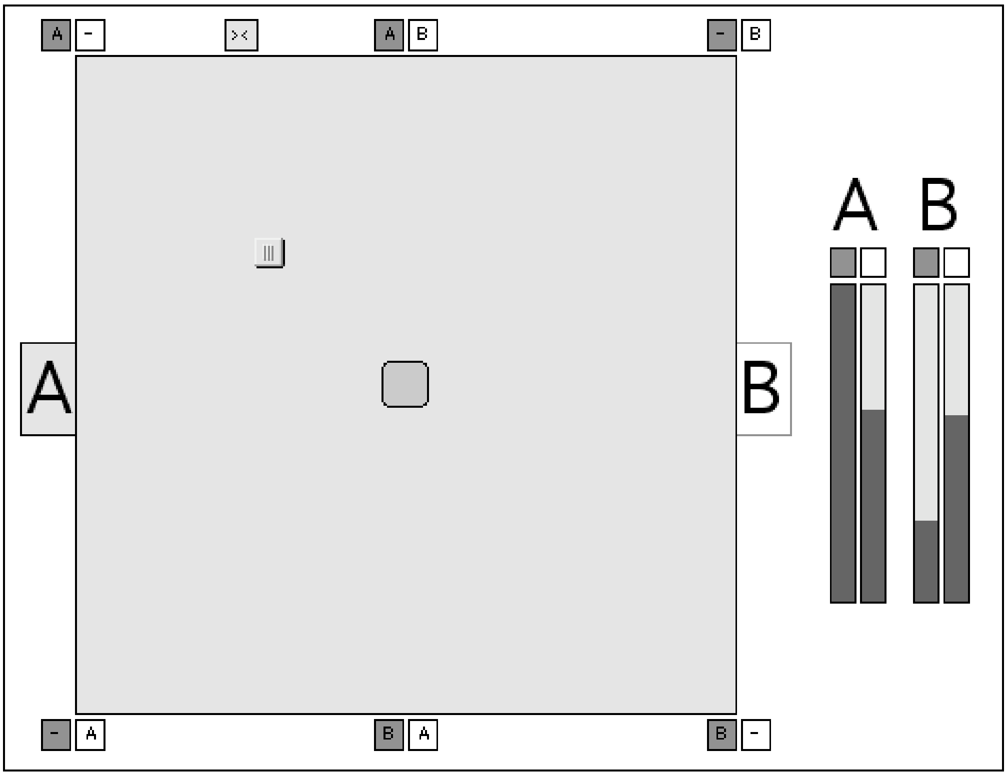

Figure 2: *GUI prototype of the 2D mixing controller and experimental level indicators showing the current mixing of the two incoming audio tracks A and B to the selected (dark grey) and unselected (white) matrix regions. The large fields A and B left and right to the square of the controller are buttons to choose the automatic matrix selection by real-time analysis.*

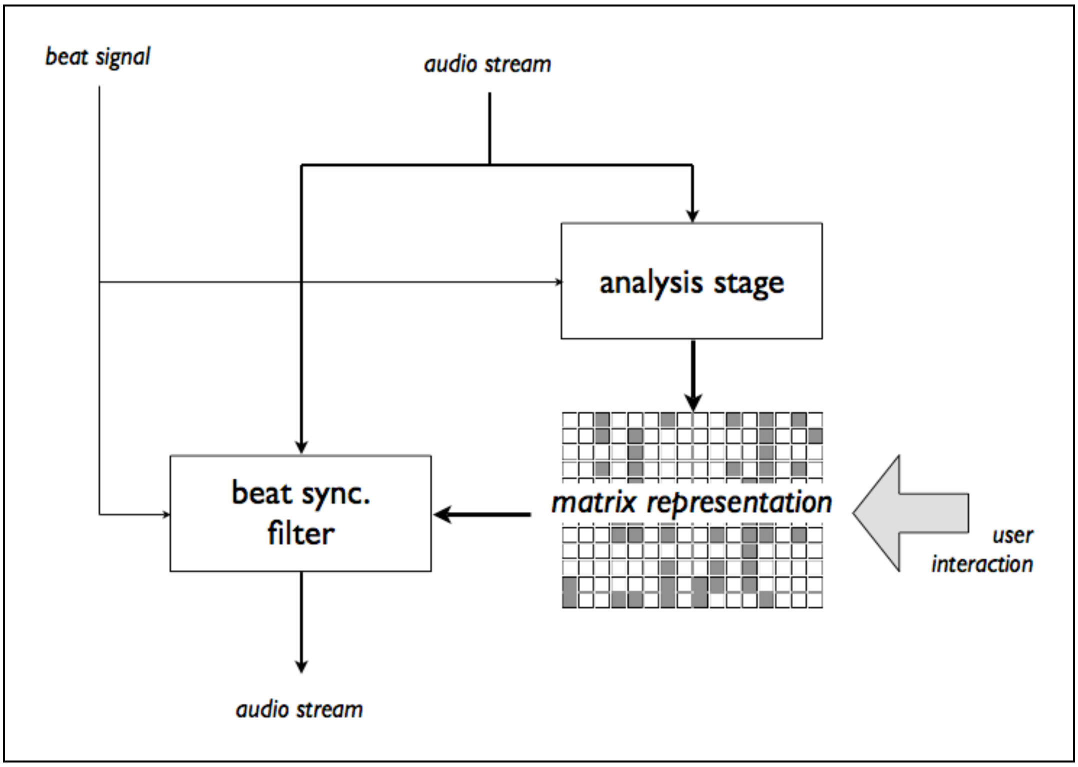

Figure 3: *Schematic overview of the real-time analysis and filter processing for each audio channel.*

The matrix squares can be automatically selected by a real-time audio analysis. The automatic algorithm would select the strongest components of one of the incoming audio streams. The user can select the analysed input stream. The automatic selection of matrix regions can be seen as a very simplified spectrogram where time is quantized into discrete beats, frequency to 12 bands and the energy to two levels, on and off.

The interaction with the provided matrix representation is intuitive and creates a novel paradigm merging multiple widespread audio processing concepts and representations into one: the drum step sequencer, the spectrogram, the equalizer and the vocoder. The usual cross-fader is replaced by a two-dimensional controller.

2.3.2 Method description and position over state-of-the-art

The X-Micks interface mainly consists of a 12 x *B* matrix representing the distribution of energy in twelve frequency bands and *B* beats (or sub-beats – typically 16th notes) of one bar of music. For music based on a 4/4 metre, a 12 x 16 matrix is displayed. The representation can be seen as a reduced sonogram obtained by coarse discretisation of the time axis to (sub-)beats and the frequency axis to perceptive frequency bands and was first developed for a simple application called Ma-Tricks. Figure 4 shows a screenshot of the Ma-Tricks sonogram of one bar –sixteen sixteenth beats– of the chorus of the song "Die Another Day" by Madonna. One can easily distinguish the bass drum and snare drum beats, without listening to the music. Watching the representation while listening to the music permits to easily associate further perceived events in the auditory stream such as further percussion instruments, but also sung words, or specific elements of the orchestration to regions in the time-frequency plane represented by the matrix squares. For example on the 11th and 12th sub-beat –just before the second snare drum off-beat on the 13th– one can clearly distinguish in the upper frequencies a column of 4 squares followed by 2, belonging to the word "yes" pronounced by the singer. The Ma-Tricks display inherits properties from two familiar representations: the sonogram and the drum step sequencer. Both representations are united to a novel intuitive representation easy to calculate from an audio stream under the condition that tempo and/or beats are known or extracted from the audio.

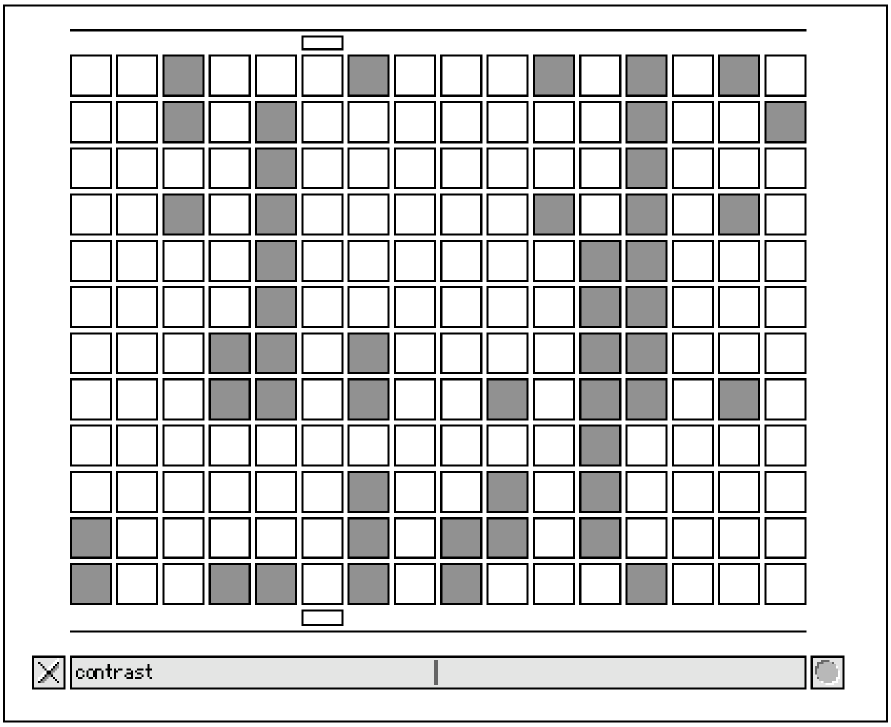

Figure 4: ***GUI prototype of the matrix display with selected (dark grey) and unselected (white) time-frequency regions. The selection shown in the example is automatically generated from a popular music track and shows the strongest frequency components of the current beat pattern.***

2.3.2.1 Perceptive frequency bands

Since the works of Eberhard Zwicker [1] various approaches have been proposed to partition the bandwidth of human hearing into perceptually equal frequency bands from different points of view. Recently real-time audio processing applications of filter banks modeling the human auditory system have been proposed to the electronic music community by Pessnitzer and Gnansia [2]. The Bark scale [3] proposes a scale of 24 critical bands of hearing up to 15.5kHz. Using the Bark scale is was easy to partition the frequency axis of the matrix display to 12 perceptually pertinent frequency bands each two Bark large. The number 12 seemed to be a good choice in terms of providing a good visual divisibility for the matrix display. An efficient implementation has been obtained by summing the corresponding frequency bins of the energy spectrum obtained by a short-time Fourier transform (SFFT). Here the FFT size has to be adjusted to permit the representation of the lowest frequency band by at least one bin and also a good approximation of the other frequency bands by the bin frequencies.

2.3.2.2 Beat-aligned analysis

Evidently the Ma-Tricks representation most pertinently applies to music with a rhythmic structure characterized by the recurrence of a limited number of beat patterns such as today's dance music. To obtain the representation, the SFFT of the 12 frequency bands has to be well synchronised to the beat of the audio stream. The energy regarding each frequency band has to be integrated over one sub-beat. If taking into account tempos between 60 and 200 bpm with a subdivision of 4 sub-beats per beat, a sub-beat has a period between 75 and 250 milliseconds corresponding to 3307.5 and 11025 samples at a sampling rate of 44.1kHz. Two possibilities have been considered to obtain the given time-frequency matrix using SFFT computation:

- Adapting FFT size and hop size to the beat
- Using a constant FFT and hopsize adding successive FFT frames belonging to the same beat

The advantage of the first is the perfect adaptation of the processing to the perceived rhythm and onsets. For the current implementation of the application, the second possibility has been adopted for its simplicity with success. The hop size has been chosen rather low (128 samples) in order to provide a sufficient temporal resolution. A bar by bar display of the beat pattern represented by the matrix synchronised to the audio stream in real-time requires either a prior (non real-time) analysis of the reduced sonogram or the delaying of the audio stream by the duration of one bar. The real-time display chosen for the X-Micks application continuously shows the analysis result of the last (sub-)beat in a column of the matrix, while the column representing the current (sub-)beat is hidden by a cursor advancing synchronously with the timing of the music. A memory effect can be added to the display reinforcing the visual presence of recurrent elements of the beat patterns of successive bars. In the context of the described work the beat analysis and tracking of the audio stream is considered calculated outside of framework of the described application. The processing requires a simple control signal (or event) giving a count for each sub-beat of a bar. Generally, audio plug-in standards such as VST, AudioUnits or RTAS, provide information concerning the signature, tempo and the exact onset time of beats from which the required sub-beat synchronous signal can be easily derived. Originally the metre and beat information of plug-ins was used in the context of composition environments mixing music representations with a metric structure and audio processing. Here it allows audio effects such as a multi-delay or a chorus to be synchronised to the metre of a synthesised accompaniment. A newer generation of tools allows beat synchronous processing by extracting the tempo and onsets in real-time from an audio stream and generating the beat related plug-in information on the fly or by a previous off-line analysis of the processed audio file. The actual availability of beat information within a particular audio application depends on the implementation of the plug-in host. Figure 3 shows an overview over the described analysis stage.

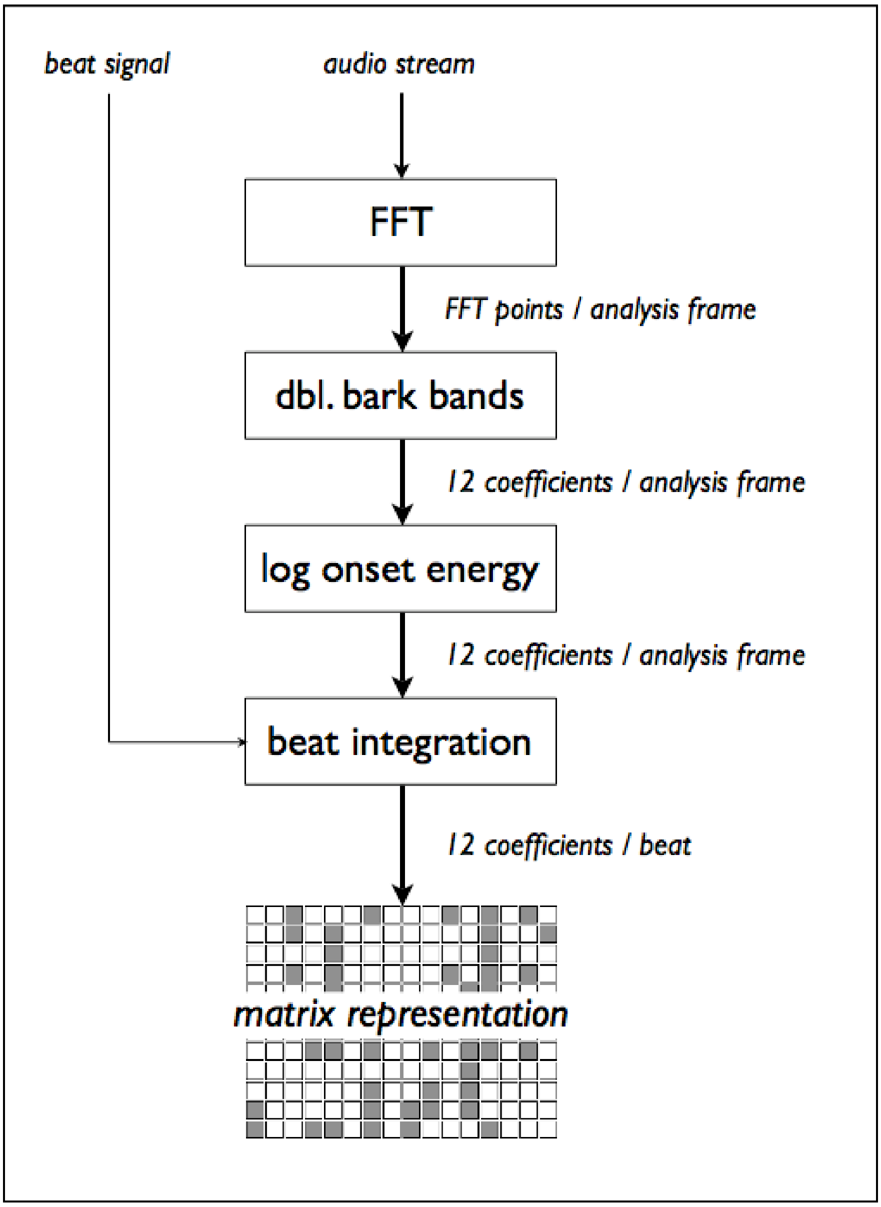

Figure 5: ***Schematic overview of the beat synchronous analysis stage. The analysis stage creates in real-time the matrix representation, which can be seen as a very simplified and quantized spectrogram of one bar of the song.***

[1] E. Zwicker, G. Flottorp, and S. S. Stevens, "Critical bandwidth in loudness summation," Journal of the Acoustical Society of America, vol. 29, pp. 548–557, 1957.

[2] D. Pressnitzer and D. Gnansia, "Pure Data," in Proceedings of the International Computer Music Conference, ICMC, Barcelona, Spain, 2005.

[3] J. O. Smith III and J. S. Abel, "Bark and ERB bilinear transform," IEEE Transactions on Speech and Audio Processing, vol. 7, no. 6, pp. 697–708, 1999.

[4] A. La Burthe and G. Peeters, "Resum¥ e¥ sonore," in Internship in the framework of the master ATIAM –final report IRCAM/INPG Grenoble, Paris, France, 2002.

[5] N. Schnell and D. Schwarz, "Gabor, Multi-Representation Real-Time Analysis/Synthesis," in Proceedings of the 8th International Conference on Digital Audio Effects, DAFx'05, Madrid, Spain, 2005.

[6] N. Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Müller, "FTM – Complex Data Structures for Max," in Proceedings of the International Computer Music Conference, ICMC, Barcelona, Spain, 2005.

[7] M. Puckette, "Combining Event and Signal Processing in the MAX Graphical Programming Environment," Computer Music Journal, vol. 15, no. 3, pp. 68–77, 1991.

[8] E. Vincent, R. Gribonval, and C. Favotte, "Performance measurement in Blind Audio Source Separation," IEEE Transactions on Speech and Audio Processing, in press.

2.3.3 Implementation

The X-Micks plugin uses the described matrix representation to interact with a real-time audio processing algorithm filtering and mixing to beat synchronised (stereo) audio streams to a hybrid audio stream. Also the beat-synchronisation of the two incoming audio streams is considered to be external to the application. In the case of a plug-in implementation the beat signal and the audio streams have to be handled synchronised to the X-Micks plug-in. This is the case for the envisaged integration of an X-Micks VST plug-in into the SemanticHifi Authoring tool host application.

2.3.3.1 The graphical user interface

Each of the two matrices of the X-Micks interface is associated to one of the input audio streams. The matrices display each the reduced sonogram of one bar of the beat synchronised audio streams in real-time. Figure 4 shows the prototype interface of the X-Micks applications, Figure 1 the final module interface.

The user can interact with the matrix display by clicking to the squares of the matrix toggling the state between unselected (grey) and selected (coloured). With the default settings, the selected matrix regions are audible, while the unselected are not. This way the user can reduce each of audio streams to certain beats and frequency bands. Short cuts are provided to control entire lines—corresponding to particular frequency bands—and entire columns—corresponding to particular beats. An additional short cut permits to select all matrix squares belonging to the same energy range in the column around a selected square and to the same frequency bands in other beats. The tolerance of the energy range around the value of the selected square can be dynamically adjusted by dragging the mouse after clicking, so that the user can select easily the time-frequency regions of distinguishable recurrent components of the audio stream such as bass drum, snare drum or hi-hat. This interaction depends directly on the analysed energy so that one could speak here of a simple case of content based interaction. The range sliders under the matrix display can be used to adjust the actual level for the selected (upper part of the slider) and unselected (lower part of the slider) time-frequency regions. The slider in the middle left of figure 3 shows that the selected regions of the audio stream represented by the left matrix is slightly lowered while the unselected regions are completely suppressed. In the contrary, the unselected regions of the audio stream represented by the matrix are not completely supressed as shown by the slider on the right.

2.3.3.2 Beat-synchronous filtering

The actual sound processing to create the final hybrid audio stream out of the two incoming audio streams is sufficiently described as time-variant beat-synchronous filtering. As well the filtering can be relatively easily implemented based on SFFT. Figure 3 shows an overview over the involved data-flow including the user interaction.

A straight forward implementation derives the reduced sonogram representation from the same SFFT spectra of the input streams which are used for the convolution and re-synthesis (by IFFT and overlap-add) after the superposition of the spectra from both streams to a single output steam. One can see the overall algorithm as a simple real-time analysis/synthesis process which provides an intermediate representation allowing for intuitive user interaction. In the case of using beat synchronous SFFT the algorithm implements a beat-synchronous overlap-add (BSOLA) algorithm as proposed by Peeters for a different context [4]. In analogy to frequency-domain PSOLA, here one could speak of FD BSOLA.

2.3.3.3 Prototyping and implementation environment

The implementation of the developed X-Micks application is entirely based on the Gabor [5] library developed at IRCAM with the FTM [6] extension for the Max/MSP [7] environment5. The matrix representations as well as the SFFT processing are implemented using the FTM *fmat* class and the related functionalities allowing very rapid and efficient prototyping within Max/MSP. The plug-in version has been developed using the *pluggo* extension for Max/MSP.

2.3.4 Dissemination materials

2.3.4.1 Scientific publications

[1] N. Schnell and D. Schwarz, "Gabor, Multi-Representation Real-Time Analysis/Synthesis," in Proceedings of the 8th International Conference on Digital Audio Effects, DAFx'05, Madrid, Spain, 2005.

[2] N. Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Müller, "FTM – Complex Data Structures for Max," in Proceedings of the International Computer Music Conference, ICMC, Barcelona, Spain, 2005.

[3] N. Schnell, D. Schwarz, and R. Müller, "X-Micks – Interactive Content Based Real-Time Audio Processing", in Proceedings of the 9th International Conference on Digital Audio Effects, DAFx'06, Montrel, Canada, 2006.

3 Dissemination materials

3.1 Public presentations

Fairs at which Traktor DJ Studio 3 was presented (as it includes features developed within SemanticHIFI, but those did not mention explicitly the IST project because they were end users oriented)

- 119th AES, New York, NY, USA - October 7-10
- Winter NAMM 2006
- WMC/Remixhotel Miami 2006

- Musik Messe Frankfurt 2006

3.2 Press articles and interviews

Reviews and Articels covering Traktor DJ Studio 3 (as it includes features developed within SemanticHIFI, but those did not mention explicitly the IST project because they were end users oriented):

- **Germany**: Amazona_03-2006, Beat_01-2006, Debug_01-2006, Groove_12-2005, Intro_12-2005, Keyboards_02-2006, Keys_02-2006, MacUp_05-2006, Macwelt_03-2006, PC&Musik_04-2006, Raveline_01-2006, Smag-11-2005 ;

- **United Kingdom**: IDJ_01-2005, ComputerMusic_02-2006, Musiktech_04-2006;

- **USA**: Remix_03-2006