

Google



“Transformaciones espectrales en tiempo real para CLAM”

Hernán Ordiales



web: <http://h.ordia.com.ar>

email: h@ordia.com.ar

Resumen de trabajo

- Ordenar y mejorar algunas partes de código existente (clases)
- Red de pitch-discretization en el NE y prototipo
- Armonizador con modelo SMS y prototipo
- Red de voz disfónica
- Efecto SMS Morph en el NE
- Mejora del algoritmo del armonizador de voces
- Trabajo sobre el algoritmo de time-stretch en el NE
- Frecuencia fundamental a nota MIDI en el NE y red de Audio2MIDI

Google Summer of Code

- Programa organizado por Google
- Sponsorea la realización de proyectos open-source
- Orientado a estudiantes
- Se realiza desde el 2005
- 900 estudiantes ; 1500 mentores ; 130 organizaciones ; 90 países

CLAM - ¿Qué es?



- Biblioteca de software para la investigación y desarrollo sobre audio y música
- Ofrece un modelo conceptual y herramientas para el análisis, la síntesis y el procesamiento de señales de audio
- Premio ACM: Mejor software multimedia open-source en el 2006

CLAM - Características

- Dos modos diferentes de trabajo:
 - Framework
 - Prototipado rápido de aplicaciones
- Software Libre (Licenciado bajo GPL)
- Escrito en C++ orientado a objetos
- Multiplataforma (Linux, Mac, Windows)

CLAM – Aplicaciones principales

- NetworkEditor
- Annotator
- SMSTools
- Voice2MIDI

NetworkEditor - Características

- Programación visual
- Similar a PureData, MaxMSP, Reaktor
- Redes en **tiempo real** u offline
- Diferentes backends
 - VST
 - Jack
 - LADSPA
 - Portaudio

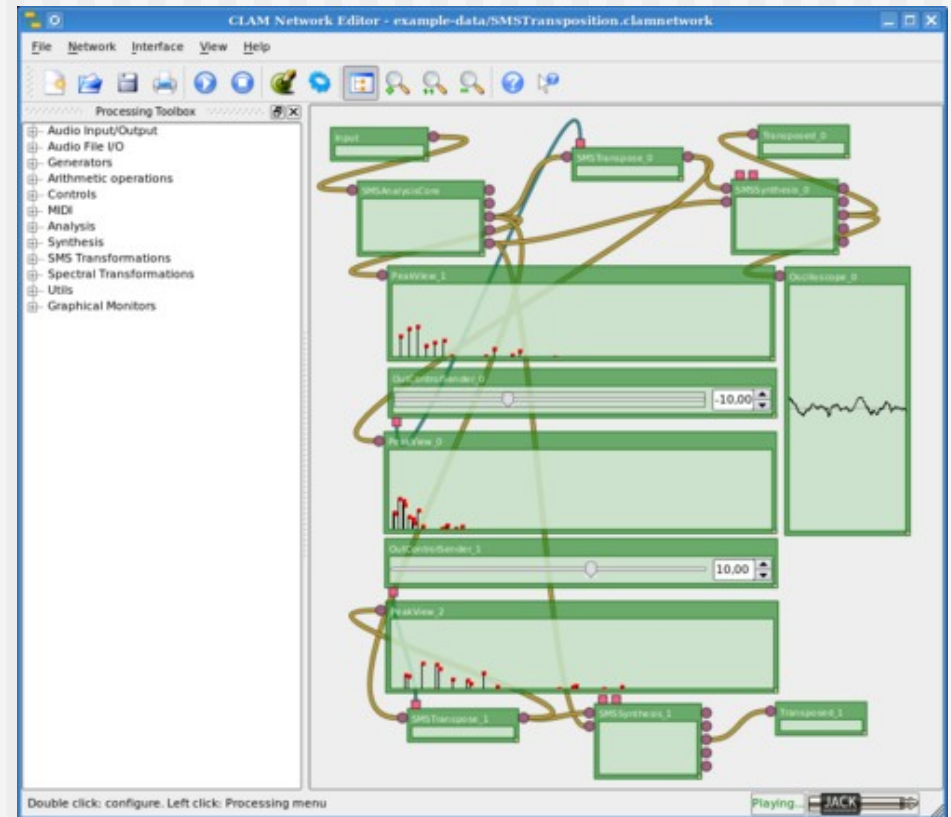


Figura 1: NetworkEditor

CLAM – NetworkEditor

- Componentes -> Processings (“cajitas”)
 - Ports (flujo de datos)
 - Controls (datos de control)
- Modelos de análisis/síntesis
 - STFT
 - Sinusoides + residuo (SMS)
 - etc
- Visualización en tiempo real

Prototipos

- Prototyper: Corre los prototipos a partir de redes del NetworkEditor (archivos XML)
- Widgets QT propios (asociados a los processings del NetworkEditor)

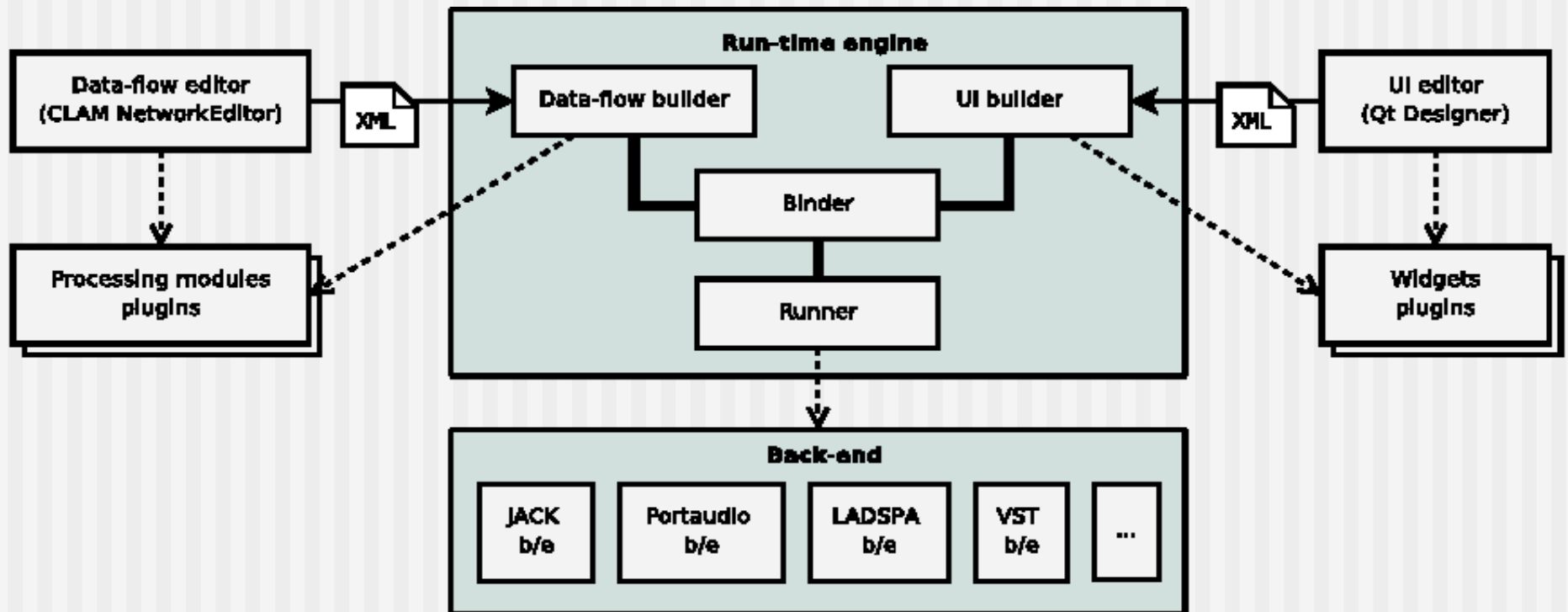


Figura 2: Esquema prototyper

Modelo sinusoides + residuo

- Modelo de análisis/síntesis para procesamiento espectral orientado a aplicaciones musicales y de audio
- Se puede ver como una generalización de la STFT y los modelos sinusoidales
- También es conocido como SMS (Spectral Modeling Synthesis) y como HILN en el contexto de MPEG4

Modelo sinusoides + residuo

- Esta modelado como la suma de un conjunto de sinusoides (los “sobretonos” estables armónicos o no, las componentes determinísticas del sonido) más el residuo de ruido (modelado como un proceso estocástico) como dos componentes separadas

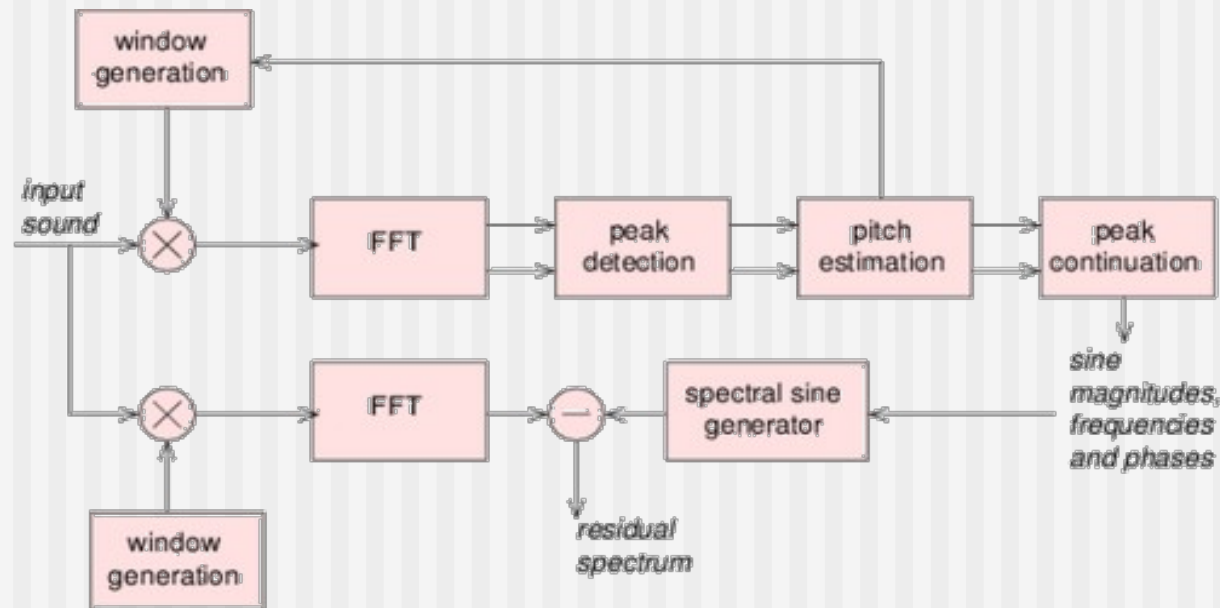
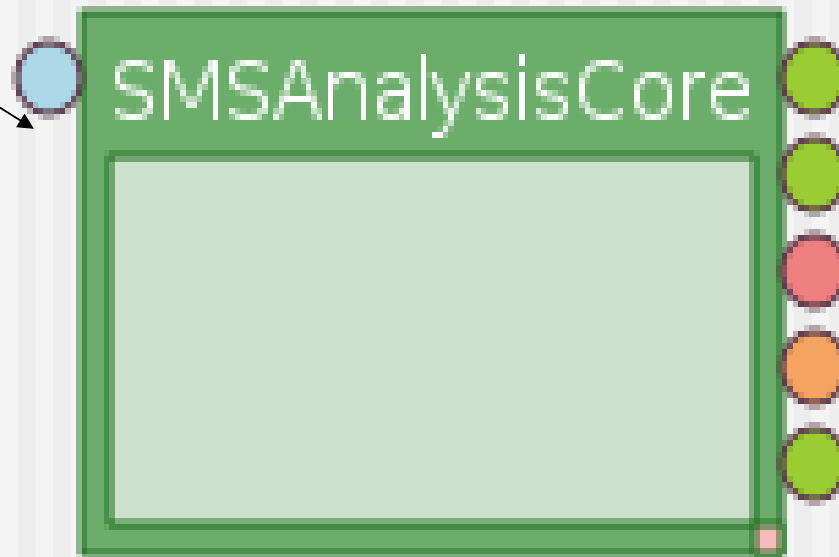


Figura 3: Esquema modelo SMS

SMS Análisis

Entrada de audio



Picos sinusoidales

Fundamental

Espectro del residuo

Figura 4: SMS Analysis Core

Morph

- A partir de dos fuentes de sonido se crea un sonido con propiedades híbridas
- Basado en interpolación (de picos y del espectro del residuo) y un balance de la fundamental (dependiente del factor de interpolación)

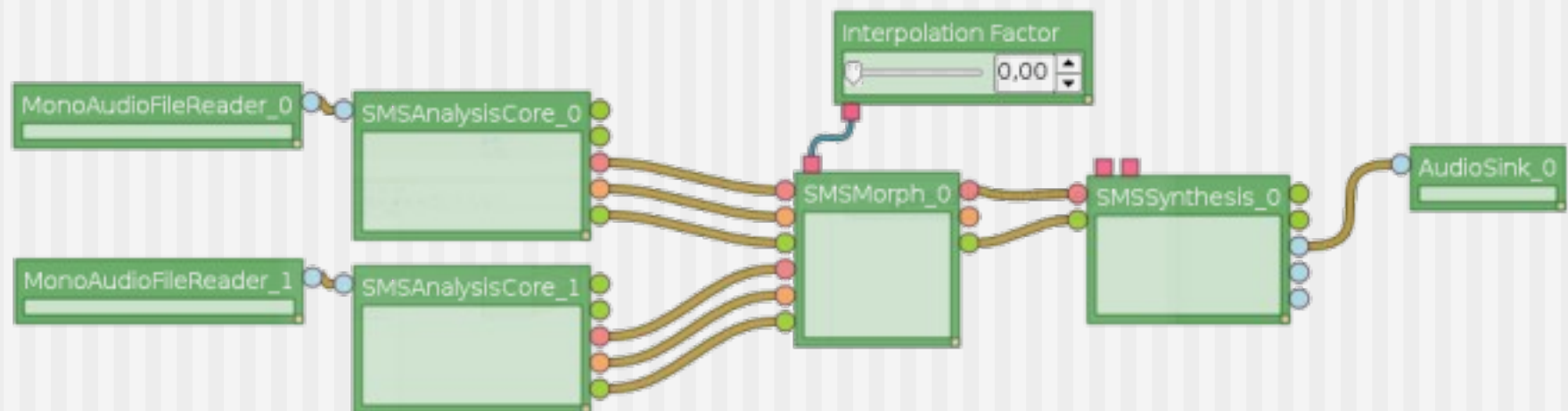


Figura 6: Red de Morph



Armonizador

- Generalmente utilizado para armonizar voces
- Dado un sonido de entrada se obtienen salidas automáticamente relacionadas armónicamente, por ej. un 3ra mayor o menor, una 5ta, una 6ta o cualquier intervalo musical que se desee
- Basado en varios SMS pitch-shiftings (uno por c/voz)
- Intervalos según escala temperada:

$$fund * 2^{\frac{p_c}{12}}$$

Armonizador

- Cantidad de voces configurable
- Parámetros (por c/voz)
 - Ganancia
 - Pitch
 - Delay
 - Detuning (pequeña desafinación para dar realismo)
- Opción de ignorar el residuo en el procesamiento (agrega mucha carga y no mejora demasiado los resultados)

Armonizador

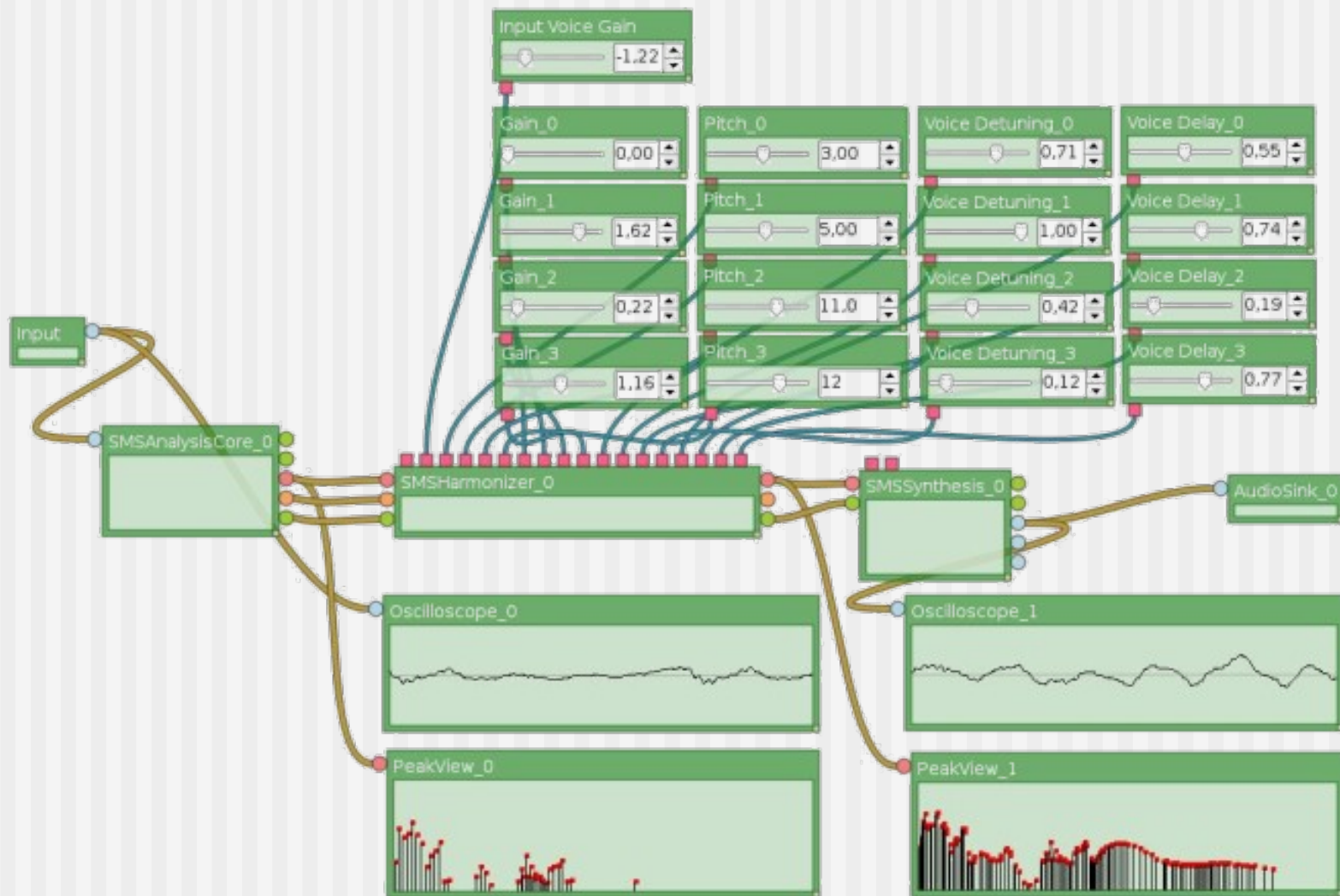


Figura 7: Red del armonizador

Armonizador - Prototipo

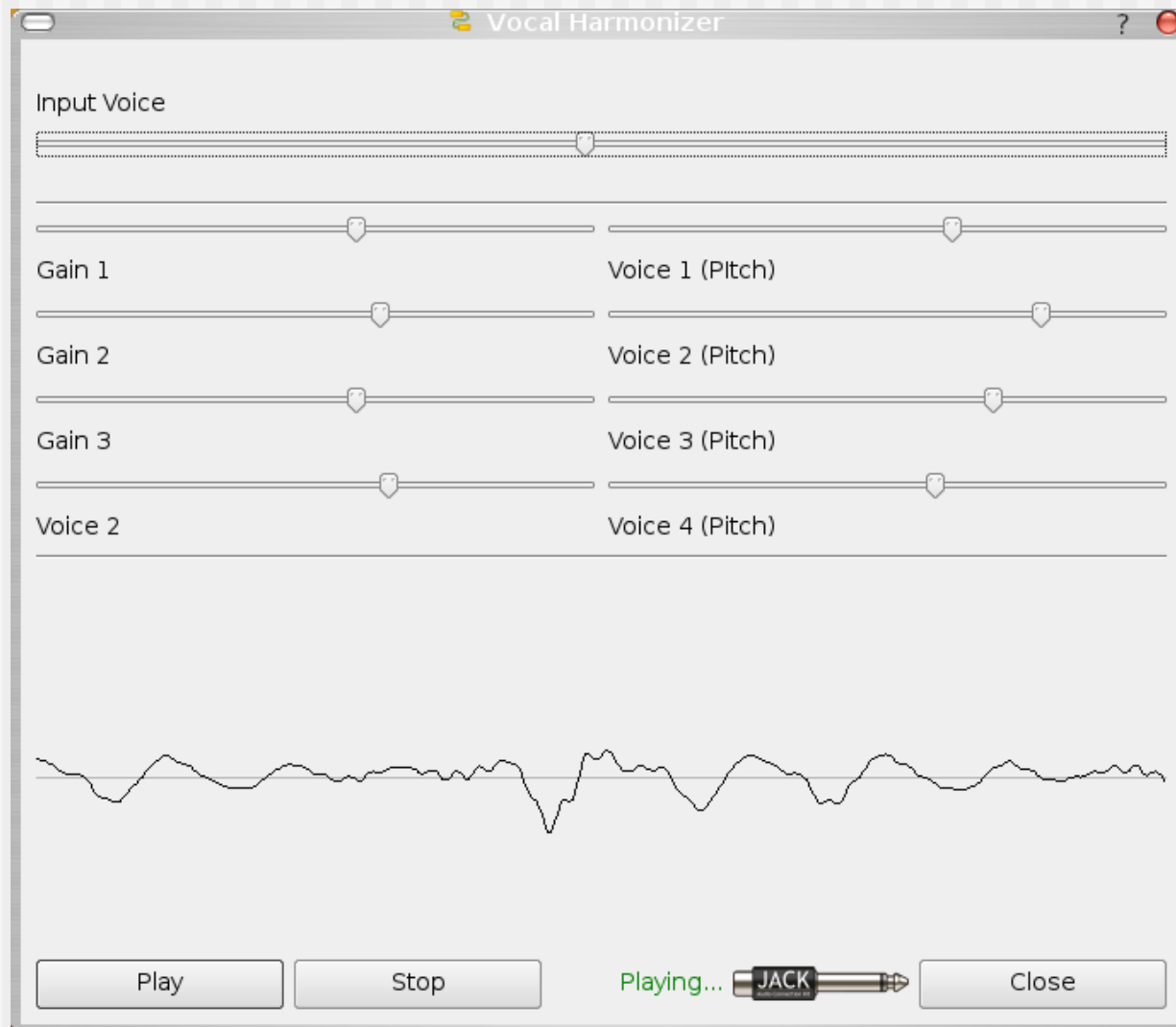


Figura 7: Prototipo del armonizador



Fundamental (Hz) a nota MIDI

$$fund_{midinote} = 69 + \log_e\left(\frac{fundfreq}{440Hz}\right) * 17.31234$$

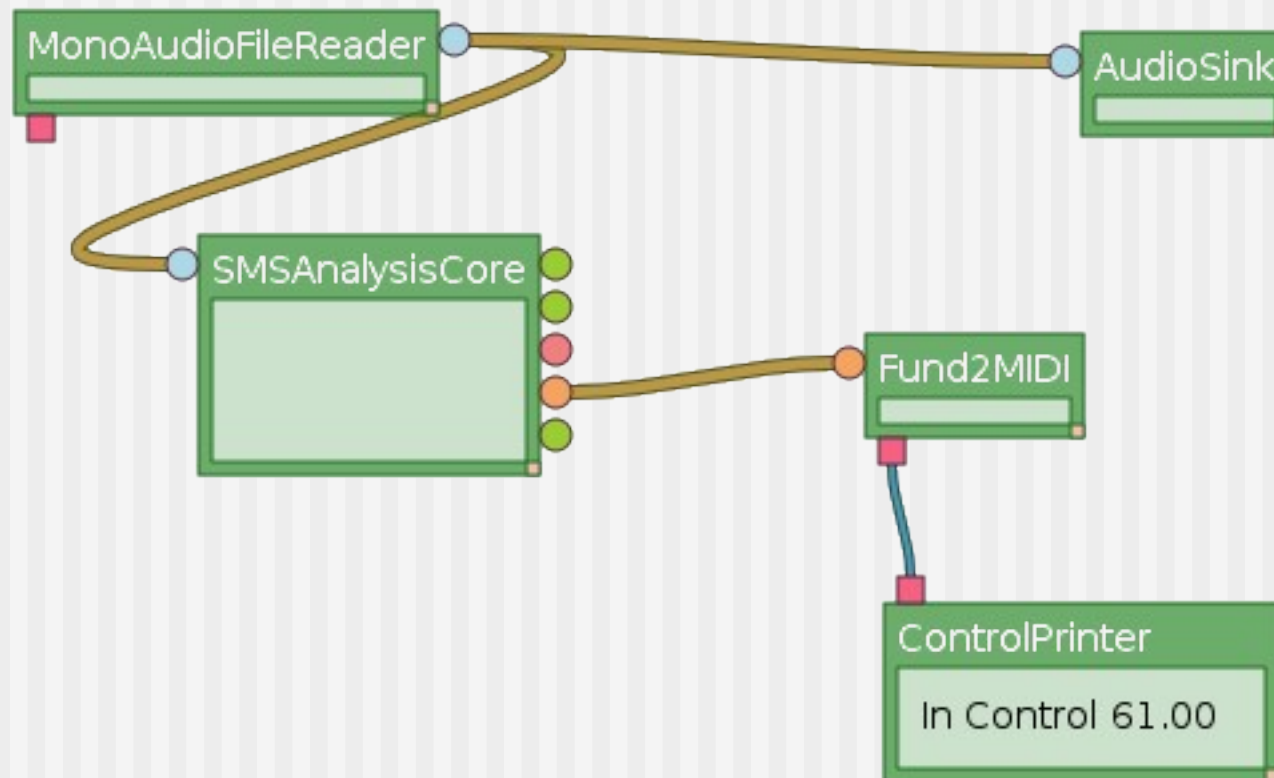


Figura 8: Frecuencia fundamental a MIDI

Organización del trabajo

- Repositorio centralizado de código (svn)
- Testfarm (sistema online)
 - Compilación multiplataforma
 - Tests de unidad, back2back, etc
- Metodologías ágiles
 - Reorganización constante de objetivos
 - Milestones (mapa de camino)
- Comunicación
 - E-mail: Lista de desarrollo y correo privado
 - Canales de chat (irc en freenode)
 - Blogs

Lo que sigue

- Mejorar el soporte MIDI en el NetworkEditor
- Mejorar algoritmos de tiempo real (NE)
- Nuevos widgets
- Nuevas redes
- Utilizar CLAM para y en otros desarrollos

Fin de la presentación

Muchas gracias!