

## TUTORIAL DE CSOUND: introducción y uso de Csound

El objetivo es conocer los rudimentos de este lenguaje para poder hacer cosas interesantes. El potencial de Csound es enorme y cubre grandes áreas relacionadas con esta asignatura, pero nuestro tiempo para él es muy limitado. El **uso tradicional de Csound** es de un compilador que se puede utilizar a través de un interfaz de usuario (como p.ej. Winsound) o mediante una orden en la consola de Linux o Mac OS X. Recientemente han surgido alternativas más cómodas y visuales para desarrollar proyectos en Csound.

### Instalación de Csound / CsoundQt

Csound puede descargarse desde su propio portal <http://csound.com> → Download, donde puede encontrarse para Windows, Mac OSX, Linux, iOS o Android. En el presente curso usaremos el CsoundQt como entorno gráfico de desarrollo, que viene incluido con las versiones más recientes del propio Csound.

En la página [https://en.flossmanuals.net/csound/\\_draft/\\_v/1.0/](https://en.flossmanuals.net/csound/_draft/_v/1.0/) se puede encontrar abundante información sobre el uso de Csound. En particular, en esa página, en la sección 02 QUICK START → A. MAKE CSOUND RUN podrás encontrar información detallada sobre el procedimiento de instalación y configuración para los diferentes sistemas operativos, tanto para ordenador de consola como para dispositivos móviles.

### Ayuda de Csound

En la web de las prácticas podrás encontrar el manual de Csound en HTML y en formato PDF en castellano (aunque ésta de una versión anticuada). La versión actual completa puedes encontrarla en <http://www.csounds.com/manual/html> y CsoundQt te ofrece esta misma ayuda integrada.

### ¿Qué es Csound?

Según se mire, Csound es un laboratorio de electroacústica, un lenguaje de programación, un entorno de diseño de sintetizadores, un procesador de audio, un sistema de composición algorítmica, etc. Csound permite programar sistemas de procesamiento y síntesis del sonido, entre otras cosas. Está concebido como un catálogo de más de 1000 operadores *software* (*opcodes*) con los que diseñar programas en el ámbito del sonido y la música por computador.

Desde el punto de vista técnico, la gran ventaja de Csound es su total flexibilidad para el diseño de virtualmente cualquier sistema imaginable, sólo limitada por el *hardware* del ordenador: memoria, disco duro, velocidad del procesador, etc. Desde el punto de vista didáctico, la cualidad que tiene Csound es que requiere del conocimiento teórico de lo que se está haciendo para conseguir el resultado deseado: si no sabemos lo que hacemos y por qué es difícil conseguir algo bien hecho. Eso puede ser malo desde el punto de vista de la eficiencia en el desarrollo, pero desde el punto de vista pedagógico es muy adecuado.

Csound es de código abierto, libre distribución y multiplataforma. Si no funciona en tu sistema, te bajas el código y lo compilas. Si no hace lo que quieres, editas el código para que lo haga y recompilas. Fue desarrollado por Barry Vercoe en el Massachusetts Institute of Technology (1985) a partir de lenguajes anteriores como Music 360 (1968) de IBM y Music 11 (1973) de los Bell Laboratories.

## Orquestas y partituras

### La orquesta

Es donde se especifican los procedimientos a realizar. En Csound “clásico” es un fichero con la extensión .orc, en los ficheros *Csound document* (.csd) está delimitada por los tags <CsInstruments>. Se hace en forma de “instrumentos” que son bloques de código que contienen variables, algoritmos y operadores para realizar la tarea deseada. Antes de definir los instrumentos, la orquesta tiene una cabecera en la que se establecen los parámetros globales de trabajo del sistema, como, por ejemplo los siguientes:

- **nchnls** = **número de canales**, para especificar si la salida será un sonido mono (valor 1), estéreo (2), cuadrafónico (4), etc. hasta 16. Si no se especifica nada, el sonido será mono.
- **sr** = **frecuencia de muestreo** de las señales de audio digital que se generen, que se indica en Hz. Si no se indica ninguna, el sistema usará 44.100 Hz.

- **kr** = frecuencia de control (en Hz). Indica cuántas veces por segundo se recalculan los valores de las variables de control que usemos en los instrumentos. Si no se especifica, su valor será de 4.410 Hz.
- **ksmps** = cuántas muestras se generan cada vez que se actualizan las variables de control. Su valor debe coincidir con **sr/kr**.
- **Odbfs** = define el valor máximo que puede tomar cualquier señal digital que maneje el sistema.

Los instrumentos (hasta un posible total de 65.536) se identifican mediante un número (N), comienzan con la palabra **instr** y finalizan con la palabra **endin**. Entre estas dos palabras reservadas, las instrucciones se escriben una en cada línea con esta estructura genérica para los operadores:

```
salida(s)  operador  entradas_separadas_por_comas  ; posibles comentarios
```

Según el operador utilizado podremos tener ninguna, una o varias entradas y/o salidas.

Además de los operadores, Csound dispone de funciones, operadores lógicos y matemáticos (con la misma sintaxis que C), así como de control del flujo, funciones de entrada y salida, etc.

## La partitura

Se llama así a la serie de órdenes de control de los instrumentos de la orquesta. En Csound “clásico” es un fichero con la extensión .sco y en los ficheros *Csound document* (.csd) se delimita por los tags <CsScore>. Las órdenes de la partitura son una única letra seguida de sus argumentos separados por espacios o tabuladores. Los argumentos pueden ser números, cadenas (entre dobles comillas) u operaciones matemáticas (en este caso, entre corchetes).

Las órdenes más utilizadas son:

- |   |   |
|---|---|
| • <b>t</b> : control del tempo                  | • <b>i</b> : activación de instrumentos |
| • <b>f</b> : generación de tablas con funciones | • <b>e</b> : final de la partitura      |

## Uso: compilación de un programa en Csound

Tanto en Linux como en Max OS X, dispones de terminales con línea de comandos que te permitirán un uso rápido y directo de Csound. En estos casos se ejecuta Csound mediante una orden, seguida de unas opciones y unos parámetros, como los ficheros de orquesta, partitura y salida, por ejemplo.

La edición de estos ficheros puede hacerse con cualquier procesador de textos que permita guardar como texto ASCII, sin formato. Existen varios en las instalaciones Linux para este fin, como “kate”, “joe”, “vi”, “emacs” o “anjuta”, pero puedes usar el que te resulte más cómodo.

En estos sistemas recomendamos tener terminales independientes para poder editar simultáneamente orquesta y partitura, y otra para la compilación. Esta se realiza mediante una orden como la siguiente:

```
csound -W -o out.wav orquesta.orc partitura.sco
```

El parámetro **-W** es para que genere un fichero WAV (si usas **-A**, p.ej., generará un AIFF), **-o** es para indicar a continuación el fichero de salida (si pones **-o dac**, la salida irá a la tarjeta de sonido directamente, si no pones este parámetro, generará un fichero test.wav). Para conocer el catálogo de opciones, puedes teclear

```
csound -h
```

La opción **-n** hace que Csound no genere ningún sonido.

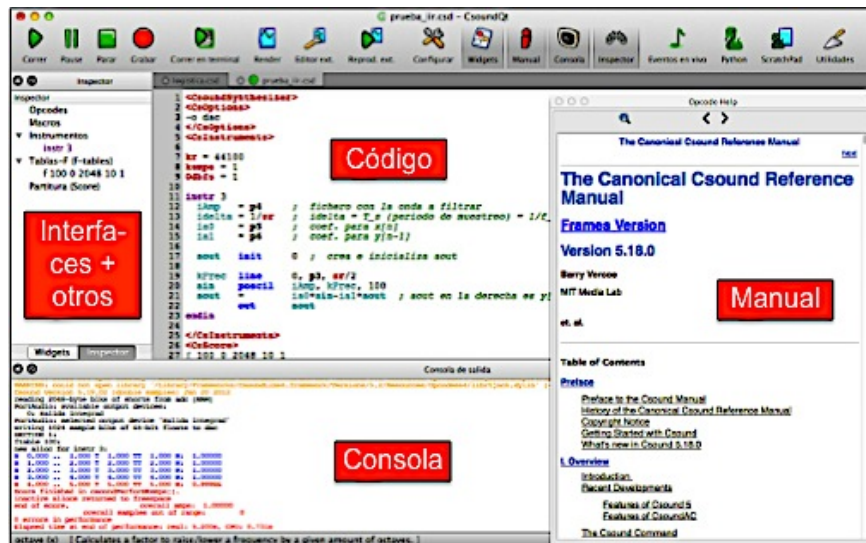
Si quieres sólo oír el resultado almacenado en el fichero de salida, puedes utilizar programas instalados como **aplay** o **play**, dependiendo del sistema, o utilizar **Audacity**, si además quieres o editar ver la onda.

Sea cual sea el sistema operativo que utilices, durante la compilación, podrás seguir el proceso de compilación mediante la consola de salida de Csound.

Sin embargo, en los últimos años han surgido *front-ends* que permiten manejar el código en Csound con una mayor comodidad y eficiencia, como Blue, Cecilia, WinXound, Cabbage, etc. Entre ellos destaca, por su implantación, el QuteCsound, que ha sido renombrado recientemente (2011) como **CsoundQt**. En 2009 se decidió incluir CsoundQt como el *front-end* estándar, incluido con la distribución de Csound, así que si instalas una versión reciente de Csound ya tendrás instalado el CsoundQt, lo cual no sucede al contrario (si te descargas e instalas CsoundQt necesitarás aparte el Csound).

Este entorno de desarrollo ofrece sintaxis resaltada, completa código y un editor gráfico para control en tiempo real. Viene con muchos programas de ejemplo, desde demos sencillas hasta complejos sintetizadores. También incluye un manual de ayuda en hipertexto con ejemplos.

A continuación se muestra un ejemplo del interfaz gráfico de **CsoundQt** (depende de la versión).



CsoundQt permite compilar el código sin necesidad de recurrir al terminal (aunque también puede hacerse) mediante botones de *run* / *pause* / *stop* y para crear ficheros de sonido mediante *Render*.

Este sistema utiliza un formato de fichero único denominado *Csound document* (.csd) que integra orquesta y partitura en un único fichero estructurado mediante etiquetas de tipo XML, que definen 3 partes:

```
<CsoundSynthesizer>
<CsOptions>
  Opciones de compilación
</CsOptions>
<CsInstruments>
  Orquesta
</CsInstruments>
<CsScore>
  Partitura
</CsScore>
</CsoundSynthesizer>
```

Estos ficheros .csd son también compilables con la línea de comando de Csound desde la versión 3.50.

Otra ventaja del uso de CsoundQt es las facilidades que da para la compilación, con su barra de herramientas, que incluye la conexión con programas exteriores para ver (y editar) el sonido generado y para reproducirlo:



Te recomendamos que utilices **Audacity** como editor externo para comprobar los resultados de tus trabajos, porque cargando en él los ficheros de audio creados, podrás ver y oír las señales generadas, pudiendo cargar simultáneamente varias, si necesitas hacer comparaciones. Además, con él podrás medir tanto tiempos de duración, como amplitudes de las ondas. Si no está así establecido en tu sistema podrás hacerlo desde **Configurar → Programas externos**.

## ¿Cómo puedo probarlo antes de entrar al laboratorio?

Lo más sencillo para probar Csound aunque no sepas hacer nada con él es abrir el CsoundQt y empezar a explorar los cientos de ejemplos de uso que hay en el menú **Ejemplos**. Los hay muy sofisticados, pero también muy simples, como los que hay en Ejemplos → Introducción → Básico o los que hay en Ejemplos → Ejemplos del manual FLOSS → ( cualquiera ). Estos últimos, además, están respaldados por una descripción técnica que puedes encontrar en [https://en.flossmanuals.net/csound/\\_draft/\\_v/1.0/](https://en.flossmanuals.net/csound/_draft/_v/1.0/)