

## **Cycling '74 has transferred some of its older website content, including this page, to PDF format.**

As a result, some of the information may be out of date. If you own the product below and are experiencing difficulties that this page doesn't cover, please contact our Support team (<https://cycling74.com/contact-support/>).

### **Helpful links:**

- <https://cycling74.com/support/challenge/>
- <https://cycling74.com/support/ilok/>
- <https://cycling74.com/support/faq/>

### **Product: FAQ: Max 4**

## **Authorization**

### **How do I authorize?**

[Visit the authorization page](#) for detailed instructions.

## **History and Background**

### **Why the names Max and MSP?**

Officially, no reason. But consider these possibilities: Max Signal Processing and Miller S. Puckette, the original author of Max. Also, it's the airport code for Minneapolis, where David Zicarelli grew up. Max is named for Max V. Mathews, computer music pioneer and sailing enthusiast.

### **Where did Max/MSP come from?**

Max was first developed for the Macintosh in the mid-1980s at IRCAM by Miller Puckette. In 1989, IRCAM started work on a real-time synthesizer card for the NeXT computer produced by Ariel called the IRCAM Signal Processing Workstation (ISPW). Puckette ported Max to the NeXT and ISPW and added a set of objects to do audio on the card. Max combined with the audio was known as Max/FTS and was widely used at IRCAM and about 30 other centers and individual studios around the world. A single card with two processors was about \$12,000, not including the computer.

Max was first released commercially by Opcode Systems, Inc. in 1990. It was chosen as Software Innovation of the Year by the readers of Keyboard Magazine in 1991. Since 1999 it has been published and supported by Cycling '74.

In 1996, Miller Puckette, now at the University of California San Diego, began developing a new program, Pd or Pure Data, aimed at the new generation of powerful microprocessors. The goals of Pd are different than Max/FTS, but it shares the ability to do real-time signal processing by connecting objects together. Initially developed on the SGI platform, Pd now runs on Windows, Linux, and Mac OSX. Shortly after Pd was started, David Zicarelli decided to add audio processing for Power PC computers to

the existing Opcode Max environment. Max/MSP is the result. It uses the Pd signal processing infrastructure, and adds features inspired by Max/FTS but not yet implemented in Pd, along with new innovations and user-interface enhancements appropriate for the Opcode Max environment.

## **How does Max/MSP compare to other audio programs currently available?**

We can divide the currently available programs into three categories: synthesizers modeled on MIDI gear, sound editors and processors, and what could be called audio rendering environments.

Native Instrument's Kontakt and MOTU's MachFive are examples of the first category. Max/MSP is clearly harder to use than these programs, which could be described as software imitations of hardware samplers. Max/MSP lets you play back samples, just as these programs do, but it is not optimized for that function. Instead it is aimed at people who are not satisfied with MIDI-style playback as a way of realizing their music, so it emphasizes real-time control and endless customization. Another comparison to make is that due to the use of specialized and computationally unchallenging algorithms, dedicated synthesizers probably have more "voices" than a completely flexible architecture such as Max/MSP. But if these "voices" are not the sounds you want to hear, that probably doesn't matter.

Max/MSP is not an audio editing program. It does all its processing in real time, and when you use it, you build a dynamic process--an instrument if you will--whereas an audio editing program is designed to make a static recording stored on your hard disk. You can certainly use the two together, taking hard-disk audio files and using them as source material for an Max/MSP real-time process, or interacting with an Max/MSP process and recording it, then processing the result further in an audio editing program.

MIT's [CSound](#) and the open source [SuperCollider](#) are programs that let you specify synthesis and signal processing algorithms using a text-based language. We could think of these programs as audio rendering environments because the computer is used to realize a description of a sound or process. The main difference between Max/MSP and these programs is ease and immediacy of use. We think Max/MSP is easier to learn than text-based programs because it offers immediate feedback and because it can be easier to conceptualize a signal processing algorithm in a graphical form than as text. It is also arguably easier to extend Max/MSP with new audio device plug-ins, abstractions and C-language signal processing externals. Both SuperCollider and CSound can be extended, but you may need to compile a new version of the entire program in order to add new capabilities. Max/MSP users have written well over a hundred signal processing external objects to date.

## **Max/MSP Macintosh-specific FAQ**

### **Leopard Support**

Max/MSP version 4.x, and all versions of our plugins are not currently supported on OS 10.5, "Leopard", but they run fine.

### **Why is "Loading CFM externals disabled" on my Intel Mac? Can I install support for these externals?**

If you are running Max/MSP 4.6 on an Intel Mac, you may have discovered an error message that says "Loading of CFM externals disabled". This means that you are trying to use older third-party externals in your patch that have not been updated for Universal Binary compatibility. Because Max/MSP is a UB

application, you will not be able to run older "CFM" externals. You will need to look for Universal Binary or Intel native updates for these objects. These can often be found on the developer's personal website, maxobjects.com, or on the Max/MSP forum.

If you are using a PowerPC Mac, you may install support for CFM externals by running the installer located inside of the Max/MSP application bundle. We provide this compatibility as a convenience to PowerPC users, however due to various changes in the application itself and the locations of various libraries, you may still need an updated external. As always, you will need to contact the developer for information and support for third-party externals.

## **What are the main differences between OS 9 and OS X versions of Max/MSP?**

Here is a brief summary of the major differences:

- OS X doesn't use OMS (Opcode MIDI System) -- the OMS objects are no longer in the 4.2 distribution. Instead, OS X handles MIDI functions using its own built-in Core MIDI; it's very much like OMS, but also includes built-in virtual sources and destinations for communicating MIDI between Max and other applications.
- The Sound Manager and the use of ASIO drivers have been replaced by OS X's built-in Core Audio, which handles audio interfacing for both built-in audio and third-party audio hardware. This means that if you're using any external hardware that used ASIO drivers, you'll need to check the manufacturer's website to download and install Core Audio equivalents to use your hardware with OS X.
- External objects are now located in the folder /Library/Application Support/Cycling '74, shared by all copies of Max/MSP: You can drop a folder into this folder and it will be included in the Max "search path" the next time you launch the application.
- Building standalone applications uses a new Max/MSP object and a simpler one-step process: Just add a copy of the new standalone object to your patch, use its Inspector to specify the options for your application, and choose Build Collective / Application from the File menu.
- There is no direct support for CD audio in Max/MSP 4.2 -- the cd and cd~ objects have been removed. Use the MSP sfplay~ object to open CD tracks as audio files instead. It's more powerful than cd or cd~, and lets you do things like playing CD tracks backwards or at different speeds.
- There is no MIDI enable/disable button in the Max 4.2 patcher window, and the lock/unlock button is the "pill" on the right side of the title bar.
- Max 4.2 can handle OS X files that have no OS 9 "type/creator" information, provided that they have the proper filename extension -- for example, files ending in .pat are Max binary files and files ending in .txt are treated as text files (Note: filenames that have those extension can be used successfully as subpatch names without including the extension in the OS X version, but not under OS 9 (version 4.1)).

## **How do the OS 9 and OS X versions of Max/MSP compare in terms of speed?**

OS X is a completely different OS than OS 9. A part of that difference involves higher system overhead. In addition, it's no longer possible for a single application to entirely take over system resources. As a result, don't expect OS X performance to be identical to OS 9 -- some things are faster, some things are slower. We're working on improving OS X performance and we hope that future updates will be even faster.

## **Will my old patches from OS 9 work in the OS X version?**

It will depend on the external objects you use in your patch. If you're using objects that are in the OS 9 distribution but not in the OS X version (such as any OMS objects), you'll have to replace them with their OS X equivalents.

If you are using third-party external objects (objects that are not part of the standard Max/MSP distribution), the OS 9 versions may not work under OS X, and may need to be replaced with OS X equivalents.

### **How will I know whether a third-party OS 9 object doesn't run under OS X?**

While many OS 9 third-party objects will run without error under OS X, you'll need to find an OS X version of an object if you see a "can't fragload" error in the Max window when you attempt to use it. In particular if the "can't fragload" error mentions InterfaceLib, MathLib, SoundLib or other OS 9 only libraries.

### **How can I find out if OS X versions of third-party objects are available?**

You can find a listing of third-party objects available for OS X [here](#), along with a report by someone who's actually checked to see what OS 9 third-party externals currently do seem to work. [maxobjects.com](http://maxobjects.com) is also a great resource for finding out about various third party objects.

### **I made a standalone application/collective for OS 9. Will I need to recompile it to run under OS X?**

Yes, you will. Be sure to check that you aren't using OMS objects or objects, objects that aren't in the Max/MSP 4.2 (or later) release, or third-party objects that do not run under OS X. Happily, making standalones is now easier.

### **If I make a patch in OS X will it work on OS 9?**

The answer is similar to that for running OS 9 patches under OS X. You'll need to replace any Core MIDI objects with OMS equivalents, check to see that you're not using any Max/MSP 4.2 (or later) objects that were not part of the standard Max/MSP OS 9 distribution, and verify that there are OS 9 equivalents to any third-party objects you use.

### **What Max/MSP 4.1 objects are not in the OS X version?**

cd, cd~, omscontrollers, omsinfo, omsnotes, and omspatches.

### **Does the OS X version take advantage of multiple processors?**

Max/MSP doesn't break audio tasks up and distribute them across multiple processors -- if you only consider the audio part of Max/MSP, there is no additional benefit to using multiple processors. Since OS X is a "threaded" OS, some applications (e.g., Logic Audio) make use of multiple processors by running some application code (e.g., MIDI and UI tasks) as one thread on one processor, and audio on another processor. While Max/MSP doesn't explicitly control how OS X distributes these threads across processors, there are some things you can do to realize some benefit from multiple processor systems.

There are three parts of Max/MSP that can potentially run in their own threads. Audio code and UI tasks

run in their own threads, so it is possible that OS X can move the audio to the second processor while the UI remains on the first. The Max/MSP scheduler can be set so that it either runs in its own thread, or run in either the audio or UI task threads using the DSP Status window (found in the Options menu):

- If "Max Scheduler in Overdrive" is off, the scheduler will run in the same thread as UI tasks.
- If "Max Scheduler in Overdrive" is on AND "Scheduler in Audio Interrupt" is on, the scheduler will run in the same thread as the audio code.
- If "Max Scheduler in Overdrive" is on and "Scheduler in Audio Interrupt" is off, the scheduler will run in its own thread.

You can use these settings to optimize the use of multiple processors, depending on what kind of UI and scheduling tasks your patch requires.

If you're running Jitter, it's important to know that Jitter processing typically takes place in the UI thread, and it seems that some Quicktime codecs support the explicit use of multiple processors to decompress frames.

### **Will one install work for both OSs on a dual boot system?**

No. You must have Max/MSP 4.1 installed under OS 9 and version 4.2 (or later) installed for OS X.

## **Max/MSP Windows-specific FAQ**

### **Is Max for Windows the same as the Mac version?**

It includes basically all of the same objects as the Mac version. Some objects in Max that are Mac-specific or no longer supported are not included in the Windows version.

### **Am I able to exchange files between the Windows and Macintosh versions?**

Max/MSP for Windows reads and writes the same binary and text format patcher and data files as the Mac version. Collectives created on Windows Max are platform independent and may be opened on either Windows or Macintosh Max version 4.3 or greater.

### **What do all the file extensions mean?**

Windows uses file extensions to tell the difference between files, so if you double click a file without a file extension, nothing will happen. You can rename your file with a .pat extension and Windows will know to use Max/MSP to open the file. Windows Max/MSP also uses .mxb for the binary patcher format, .mxe for externals, and .mxf for collectives.

### **What kind of audio support does Max/MSP have?**

Max/MSP has support for MME, DirectSound and ASIO audio drivers. It can act as a host for VST plugins and instruments. It can also act as a Rewire host and client.

### **What soundcards work with Max/MSP?**

All standard internal soundcards, and all other soundcards which support MME, DirectSound and ASIO drivers. If for some reason your soundcard is not working, even though it works with your other Audio Applications, please drop us a line or call, we want to hear about it.

Special Notes about soundcards.

- RME HDSP cards have MME and DirectSound drivers, but do not operate stably using current versions of these drivers. The HDSP system is optimized for ASIO driver support, and we recommend using this ASIO driver within Max/MSP.
- Laptop computers using the SoundMAX system should always use the MME driver, despite its relatively high latency. DirectSound and ASIO drivers are not directly available for this hardware, and will use emulation layers that will add more latency than already present.

### **Are DirectX plugins supported?**

DirectX plugins are not supported.

### **Can I use Windows Max/MSP to design plug-ins for Pluggo?**

Yes.

### **Will Max support Windows 95? 98? 2000? NT?**

Cycling'74 only officially supports the use of Windows XP, Home or Pro. It may run on Windows 2000 but is not supported. We offer no guarantees whatsoever on any other iteration of Windows.

### **What processing speed is recommended?**

Max/MSP runs well on PIII's, P4's, Athlons, and similar processors of various clock rates. A 500 Mhz PIII or better processor is recommended.

### **How much room do I need on my hard drive for Max/MSP?**

A full install of Max/MSP will use about 25 megs.

### **What's all this about Quicktime, do I really need it?**

Windows Max/MSP, as a result of its Macintosh roots, still relies on Quicktime code for parts of its internal operations. At this stage we recommend a FULL install of Quicktime, which will make possible many of Max's advanced graphic features, and mp3 support, among other things. You can download a Quicktime for Windows installer [here](#).

### **Why won't my SDII files work?**

SD2 files are inherently a Macintosh only format (resource based header). They will never work directly in Max/MSP. They will need to be converted first into something else (WAV, AIFF). However, if you were to transfer SD2 files to the PC and knew the sampling rate/bit depth/number of channels, you could open the raw file and make use of the force header calls to play back the file. The data byte offset would be zero, as the data fork contains only audio data.

## **What is the denormal problem and how does Max/MSP get around it?**

The denormal problem is the situation which arises when calculations involving extremely small numbers use a very large proportion of the CPU on x86 processors. These small numbers are most commonly the result of some system with feedback, e.g. a delay line, reverb, or IIR filter such as biquad~, allpass~, comb~, lores~,etc. The MSP filter objects and the times~ object (\*~) on windows have special code to set these small values to zero. As a precaution, if one were to use tapin~/tapout~ or send~/receive~, one should insert at least one copy of times~ (\*~) in the feedback chain to avoid these small numbers.

## **Developing Pluggo Plugins**

### **How do I develop my own plug-ins?**

You'll need Max/MSP. Pluggo plugins may be created directly with Max/MSP 4.5.4 or later.

### **Is it hard to develop my own plug-ins?**

If you have experience with Max and MSP, we think developing plug-ins will be very straightforward. Designing a good user interfaces for any software is always a challenge, so if you're not satisfied with the sequencer's default interface or the pluggo egg sliders, you might spend a lot of time tweaking. If you don't have experience with Max and MSP, they are deep programs but the basic programming concept is very simple. There are over 40 interactive tutorials that come with Max, and over 25 that come with MSP. Thousands of other people around the world have learned the software and are better people for it.

Finally, we believe that the plug-ins already included with pluggo should serve as an inspiration for what is possible to do. Most of them are far more idiosyncratic than what you'll find from other commercial developers. We believe that it's possible to use the plug-in development tools in a very personal context, in which you express certain ideas you have about sound and musical organization within the plug-in itself. Or perhaps you'll make a plug-in that you simply think of as a piece of music. If that sort of thing interests you, the Max/MSP system is the way to go...it opens a space that simply doesn't exist if you try to program DSP algorithms in C and connect interfaces to them.

### **Max has a standalone application generator that developers can use without royalties. What about plug-in developers?**

Yes - grab the Pluggo runtime [here](#).

### **Pluggo runs on both Windows and Macintosh systems. Does this mean that I can create plug-ins that will run on both Windows machines and Mac using Max/MSP?**

You create Pluggo plug-ins from Max/MSP patches, so you can make a plug-in by creating it on a Macintosh or Windows machine, and then saving your patch as a plug-in. If you save the patch as a plug-in on a Windows system, you'll get a plug-in that runs on either VST or RTAS host applications on a Windows machine. If you save the patch as a plug-in on a Macintosh, you'll get a plug-in that runs on VST, RTAS or Audio Units host applications (Please note that you can't take a plug-in created on one operating system and run it on the other).

### **Do I have to do anything special to develop plug-ins for different host applications?**

No. When you save a Max/MSP patch as a plug-in, the plug-in will run in any format supported on the operating system (VST/RTAS on Windows, AU/VST/RTAS on the Macintosh)-you don't need to do anything special.

## **Max/MSP Operation and Compatibility**

### **Where is the processing done?**

Directly on the computer's CPU. The processing occurs at interrupt level in the background so you can use your computer for other things at the same time, although if you do too much processing, the background begins to intrude rather severely on the foreground. There has been a tremendous increase in the speed of processor chips in the past year or two, and this is responsible for the ability for programs such as Max/MSP to perform so well in real time.

### **Why can't I download a patch onto my DSP card?**

MSP's signal processing architecture is specific to general purpose computers.

### **Does Max/MSP take advantage of multiprocessors?**

Max/MSP doesn't break audio tasks up and distribute them across multiple processors -- if you only consider the audio part of Max/MSP, there is no additional benefit to using multiple processors. Since OS X and Windows XP are "threaded" OSs, some applications (e.g., Logic Audio) make use of multiple processors by running some application code (e.g., MIDI and UI tasks) as one thread on one processor, and audio on another processor. While Max/MSP doesn't explicitly control how OS X distributes these threads across processors, there are some things you can do to realize some benefit from multiple processor systems.

There are three parts of Max/MSP that can potentially run in their own threads. Audio code and UI tasks run in their own threads, so it is possible that OS X or XP can move the audio to the second processor while the UI remains on the first. The Max/MSP scheduler can be set so that it either runs in its own thread, or run in either the audio or UI task threads using the DSP Status window (found in the Options menu):

- If "Max Scheduler in Overdrive" is off, the scheduler will run in the same thread as UI tasks.
- If "Max Scheduler in Overdrive" is on AND "Scheduler in Audio Interrupt" is on, the scheduler will run in the same thread as the audio code.
- If "Max Scheduler in Overdrive" is on and "Scheduler in Audio Interrupt" is off, the scheduler will run in its own thread.

You can use these settings to optimize the use of multiple processors, depending on what kind of UI and scheduling tasks your patch requires.

If you're running Jitter, it's important to know that Jitter processing typically takes place in the UI thread, and it seems that some QuickTime codecs support the explicit use of multiple processors to decompress frames.

### **What kind of performance can you get?**

Because Max/MSP is a real-time program, the speed of your processor imposes a hard limit on what you can do. If you are serious about doing audio signal processing on your computer, you want the fastest machine you can buy. With a real-time program, running out of processing capacity means that the sound begins to distort, or your computer is extremely unresponsive, or both. We have seen dramatic improvements in processor speed recently, but once you start using Max/MSP, you may want even more.

We allow you to download Max/MSP and try it on your own computer because we don't want to misrepresent its performance. Its value to you is inherently tied to what it can do on the computer you own. If you try it and subsequently want to upgrade your computer or get a new one, well it's refreshing, since most people think that Photoshop is the only reason to buy a fast computer. You'll know otherwise.

### **Can I use a DSP algorithm I develop with Max/MSP in another audio program such as Cubase or Digital Performer or Pro Tools?**

Yes. MSP contains a set of objects you need for building pluggo audio plug-ins for VST, MAS, or RTAS applications. You don't need to purchase pluggo in order to create your own audio plug-ins, and the [pluggo runtime installer](#) allows you to share your audio plug-ins with everyone -- even people who don't own pluggo.

### **Can I pipe a software synth into Max/MSP?**

Max/MSP can receive data from ReWire instruments, and VST hosts.

### **Does Max/MSP support plug-ins for other audio programs?**

Yes. Max/MSP contains an object called vst~ which lets you load a VstPlugin. Plug-ins in Max/MSP can be more interesting than the way sequencers use these plug-ins. For instance, you could load ten plug-ins and assuming you had the CPU power to do it, run them all in parallel. Or, using Max, you could generate algorithmic automation control over several plug-in parameters at once. You can also use VST instrument plug-ins and send them MIDI events, as well as ReWire devices.

### **What are the advantages of using an audio interface card with Max/MSP?**

There are several advantages. The first is audio quality. Most I/O cards available on the market are 24-bit, while the OS 9 Sound Manager and OS X Core Audio is 16-bit. Since Max/MSP's internal processing uses 32-bit floating point numbers, there is definitely a loss in output resolution at the DAC. Most cards use digital inputs and outputs, eliminating CPU interference that reduces the signal-to-noise ratio obtainable from analog outputs that are touching the computer hardware.

Another advantage is latency. The design of the Sound Manager and the Macintosh audio hardware makes it difficult to achieve suitable latency performance. Using an audio card that Max/MSP supports directly allows you to bypass the Sound Manager completely.

Max/MSP 4.1 for OS 9 supports ASIO-compatible audio cards and interfaces; virtually all products advertised as Mac-compatible will work with ASIO.

Max/MSP 4.2 for OS X supports Core Audio. You'll need a Core Audio driver for your hardware to use it with OS X.

All version of Windows Max/MSP support ASIO, directX and MME drivers, as well as ReWire.

### **Does Max/MSP work with Digidesign's Pro Tools hardware?**

Yes, it does. If you want to use ProTools hardware such as an Mbox or Digi 001 with OS 9, you'll need to put the Digidesign DirectIO ASIO driver in your ASIO drivers folder. If you're using OS X, you'll need the Digidesign CoreAudio driver.

On Windows, Max/MSP uses the Digidesign ASIO driver.

### **What type of sound files does Max/MSP support? Can I record what I do in Max/MSP to a file?**

Max/MSP reads and writes AIFF, Sound Designer II, WAVE, NeXT, and raw data files, and lets you import any audio file format supported by QuickTime. You can connect one or more audio signals to a hard-disk recording object in Max/MSP and record whatever you are synthesizing or processing. The file can contain up to 16 channels, and may be saved in a variety of sample formats from 8 to 64 bits. Additionally the buffer~ object can import the audio track from anything that is supported by QuickTime (e.g. .mp3, .mov, .avi, etc.).

### **I see that Max/MSP has delay line objects. How long can I make a delay?**

Delay lines are stored in the memory of the computer, so the maximum delay time is limited by the amount of memory you have available. A mono delay uses 1MB of memory for each 5.94 seconds of audio. So, if you used 32MB for a delay line, you would have a delay line of more than three minutes (190 seconds). You can have many taps off the same delay line without using additional memory.

## **Max/MSP Technical Questions**

### **How do you keep MSP from loading when Max starts up, if you don't want to use it for that session?**

With MSP 2, you need to move all audio objects out of your max-startup folder before launching Max/MSP. We suggest making a folder inside the max-startup folder called disabled and putting the objects in there. The audio objects you have to move are: ezadc~, ezdac~, filtergraph~, gain~, meter~, and scope~. MSP is still "potentially" active however. If you load an audio object or open the DSP Status window, MSP will start up and attempt to initialize the currently selected audio driver.

### **I have a Max/MSP patch that makes a nice sound when I play a MIDI note. How do I make it polyphonic?**

You'll want to put your patch inside a poly~ object. Refer to the poly~ tutorial in the MSP Manual for an example.

### **On my somewhat ancient machine, I get frequent clicks and dropouts when recording using sfrecord~. What can I do?**

Try increasing the sfrecord~ object's buffer size. This is done by supplying a second argument to the

object after the number of channels you want to record. Try a size of 80640. The default size is 40320. This advice also applies to `sfplay~` if you get clicks or dropouts when playing back a sound file from disk.

### **If I make a patch with feedback (where an object's output is eventually routed back to its input), there is no audio output. How can I make some feedback, man?**

There are two ways, man. One way is to use `tapin~` and `tapout~` with a delay of 0. There will be a delay equal to the current signal vector size (the minimum is 2 samples in MSP 2 and either 32 or 64 samples in MSP 1.x). The other is to send the signal through a `send~/receive~` pair. The delay is the same as in the `tapin~/tapout~` case.

### **Your oscillators aren't accurate. Most of the time they never reach exactly 1. Why not?**

A digital oscillator is a discrete representation of a continuous function. The function is sampled at an interval determined by the sampling rate. If the function reaches a value of 1 only between two samples, you won't see this value in the sampled output.

### **What's the best version of the Macintosh and Windows OS to use with Max/MSP?**

Versions of Max/MSP up to 4.1 require OS 8.1 or later. Max/MSP 4.2 requires OS X version 10.2 or later. Max/MSP for Windows requires Windows XP.

### **Is there any way that I can use QuickTime Instruments with Max/MSP or use something like a softsynth without using VST?**

As of Max/MSP 4.5, standard synthesiser playback devices are provided as selectable MIDI output ports. On Windows, this is typically the General MIDI synthesizer that comes with your sound card. On OS X, this is Apple's DLS Synthesizer.

## **Beginner's FAQ**

### **I want to look at a patch someone posted to the list, but all I see is this text stuff. How do I turn this into a patch?**

Sending Max/MSP patches as text is often the most reliable and convenient way to transport them via email and even using jump drives. Often, patches saved in binary form will get corrupted in the process of transporting them. For this reason, it is customary in the Max community to post patches as text.

To convert a text into a patch:

- Select the text you wish to copy from the email/forum.
- Open Max/MSP
- Select File->New->Patcher
- Paste. Voila!

See also: "WhatsNewInMax455", pg. 13

## **I'm having trouble with Max/MSP and need some help. How can I get the assistance I need?**

- Check the documentation. Often, your questions will be answered by a quick search through the PDF files.
- If you are having problems with general functionality, please send an email to [support@cycling74.com](mailto:support@cycling74.com).
- If you are looking for advice about a project, a troublesome patch, or other miscellaneous topics, chances are it has been covered on the forums. Doing a search of the forum archives will usually turn up a variety of answers to any question. If you don't find what you're looking for, post your question to the forum.

## **I would like to post my question to the forum. Any suggestions?**

- Before posting a question, it is advised that you search the forum archives and product documentation first. This will help you to avoid asking a question that has been answered many times before.
- Try to be as clear and detailed as possible. Usually, including an example patch that demonstrates the question is very helpful. By taking the time to formulate your question, you will greatly increase your chances of getting it answered.
- Be courteous and respectful of the community. As with any online forum, it is advisable to spend some time "lurking" before jumping into discussions.
- If you are still having doubts, check out [How To Ask Questions the Smart Way](#) by Eric S. Raymond and Rick Moen.

## **I've got this great idea for a patch, but don't know where to start. Where can I find some helpful patches I can borrow ideas from?**

- The Forum Archives - Throughout the years, a ton of really useful patches have been posted to the list, and are usually accompanied by some thought-provoking discussion. The searchable forum archives are an amazing resource for Max users.
- The Max/MSP "examples" folder - Often overlooked, this folder is packed with clever patches and demonstrations of more advanced techniques.
- Cycling '74 Articles - We strive to add new and useful content here for users of all skill levels. Check back regularly for new additions.