Logic Pro 7.1

Personal Manual by Edgar Rothermich
EdgarRothermich@mac.com <http://homepage.mac.com/edgarrothermich>

MIDI Setup

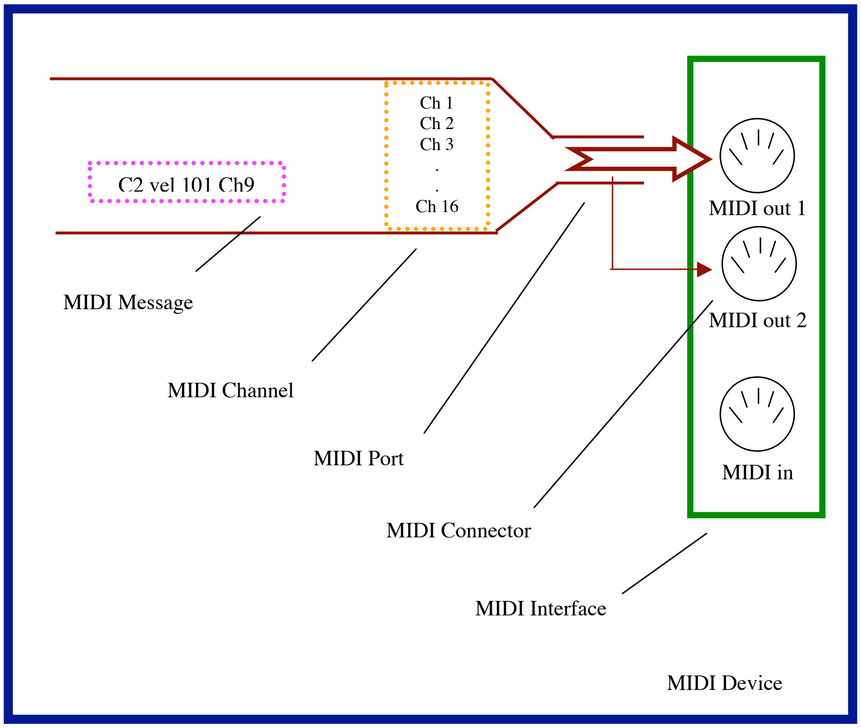
MIDI Basics

Any musical instrument has a part that initiates the sound and a part that actually makes the sound:

1) Acoustic Instrument
A piano has a key attached to a hammer (transmitter) which hits the strings (receiver) and makes the sound. The transmitter and receiver mechanism is one unit.
2) Electronic Instrument
In the old days before MIDI, you had a synthesizer with a keyboard that sends an electronic signal (transmitter) to a circuitry that creates the sound (receiver). The transmitter and receiver is again one unit.
3) MIDI Instrument
The basic idea was to separate the transmitter unit (keyboard) and the receiver unit (sound module) by creating the "MIDI Interface". This is a protocol (or a computer language) that the transmitter and receiver speak. Hitting a key on a MIDI keyboard creates a "MIDI message". Sending that MIDI message to a MIDI sound module that understands MIDI will create the appropriate sound. The transmitter and receiver don't have to be one unit. Different units can be connected via a MIDI cable that transmits the MIDI message from the transmitter to the receiver. Because the MIDI message is a digital information (1 and 0), it can be stored on a computer (Sequencer) for later transmission.

Terminologies:

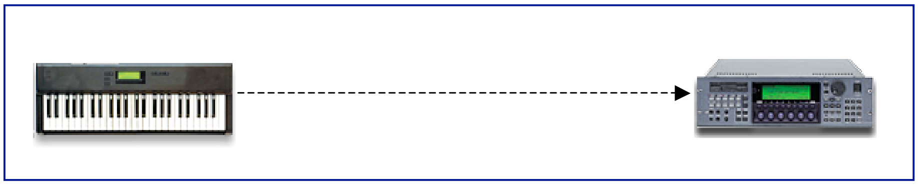
A **MIDI Device** is any electronic device with a MIDI Interface
A **MIDI Interface** is considered the part of the device with one or more MIDI connectors
A **MIDI connector** is the physical outlet (MIDI Jack) of a MIDI port
A **MIDI Port** is the electronic pipeline where the MIDI messages travel. One MIDI port can carry 16 MIDI channels.
A **MIDI Channel** is assigned to each MIDI Message.
A **MIDI Message** carries the actual information (i.e. C1, velocity 102) and is always assigned to a specific MIDI channel (1 ...16)



Please keep the terminology in mind for a better understanding of the discussion of the topics below.

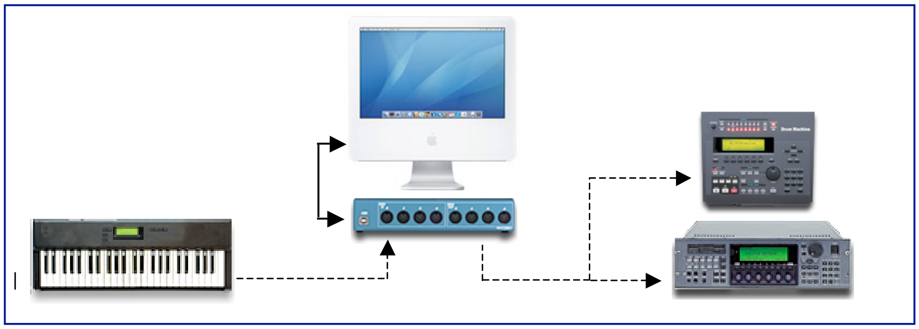
One to One MIDI Setup

Playing the keyboard generates the MIDI Message that will be send on one of the 16 MIDI channels on Port 1 which is available on the MIDI connector labeled “MIDI Out”. Connecting a MIDI cable from that “MIDI out” connector to a Sound Modules “MIDI IN”, will send that MIDI Message to the Sound Module, which plays a sound assigned to that MIDI Channel.



Computer MIDI Setup

A “MIDI interface” is the general term for that (build-in) part of a MIDI device with the MIDI jacks, which enables the sending and receiving of MIDI signals. A Computer, which you can use to run a MIDI sequencer application, doesn’t have a build-in MIDI interface. For this purpose you use an external MIDI interface, which has the MIDI jacks (up to 8 MIDI ins and outs) and a USB connector to run the MIDI messages from the Box to the Computer. Note that a MIDI cable is always one-directional where the USB connection between the MIDI interface and the computer is bi-directional. The (up to 8) MIDI in and MIDI out connectors show up on the MIDI sequencer as 8 MIDI in Ports and 8 MIDI out Ports. The Sequencer also acts as MIDI routing device where you can direct specific incoming ports to specific MIDI tracks and the output of MIDI tracks to specific out ports.
Each MIDI interface needs a driver, which is a little piece of software with basic instructions for the communication between the hardware and software.

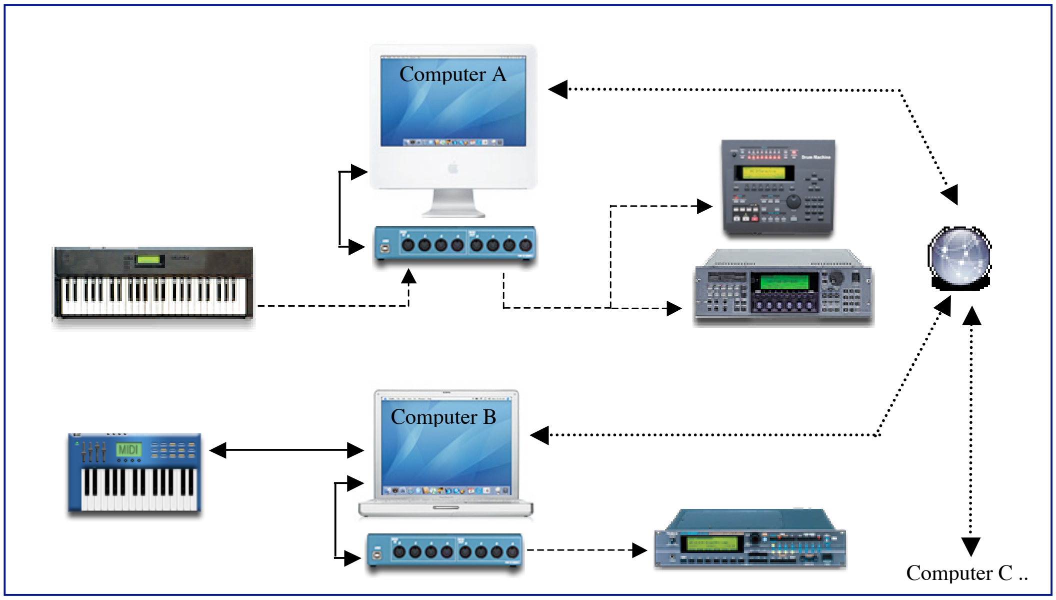


Networked Computer MIDI Setup

A more complex setup connects two or more computer systems (Computer A, B, C…) via a network. Her is the idea:

- The MIDI Computer processes all the MIDI messages (incoming and outgoing) internally
- Almost all Computers nowadays have Ethernet network capabilities and are most likely connected to a LAN or the Internet.
- Route the internal MIDI signal to the Ethernet port and send/receive MIDI over the network to and from other MIDI computers.

Now you can play your sequence you were working on your laptop (B) through the MIDI devices connected to your Studio computer (A) or you can have one or more extra computer running Logic with fully loaded EXS or audio instruments and send the MIDI via Ethernet without the need of expensive MIDI interfaces to handle the MIDI signal flow.



CoreMIDI

CoreMIDI

Once the MIDI message enters your computer via the USB cable from your MIDI interface, OSX takes over with its CoreMIDI technology. Having CoreMIDI as part of the operating systems guarantees an optimum of compatibility between different MIDI components both Hardware and Software. In the old days, OS9 didn't understand MIDI on the system level, so there were different technologies (FreeMIDI, OMS, etc), which made it, work. But different devices worked only with specific protocols and compatibility was not always guaranteed.
Every Software or Hardware developer who creates a product for OSX that handles MIDI has to make it only compatible with CoreMIDI and therefore is automatically compatible to all the other products following the same CoreMIDI implementations. The advantage for the user is a hustle-free build in MIDI compatibility in OSX.

Audio MIDI Setup (AMS)

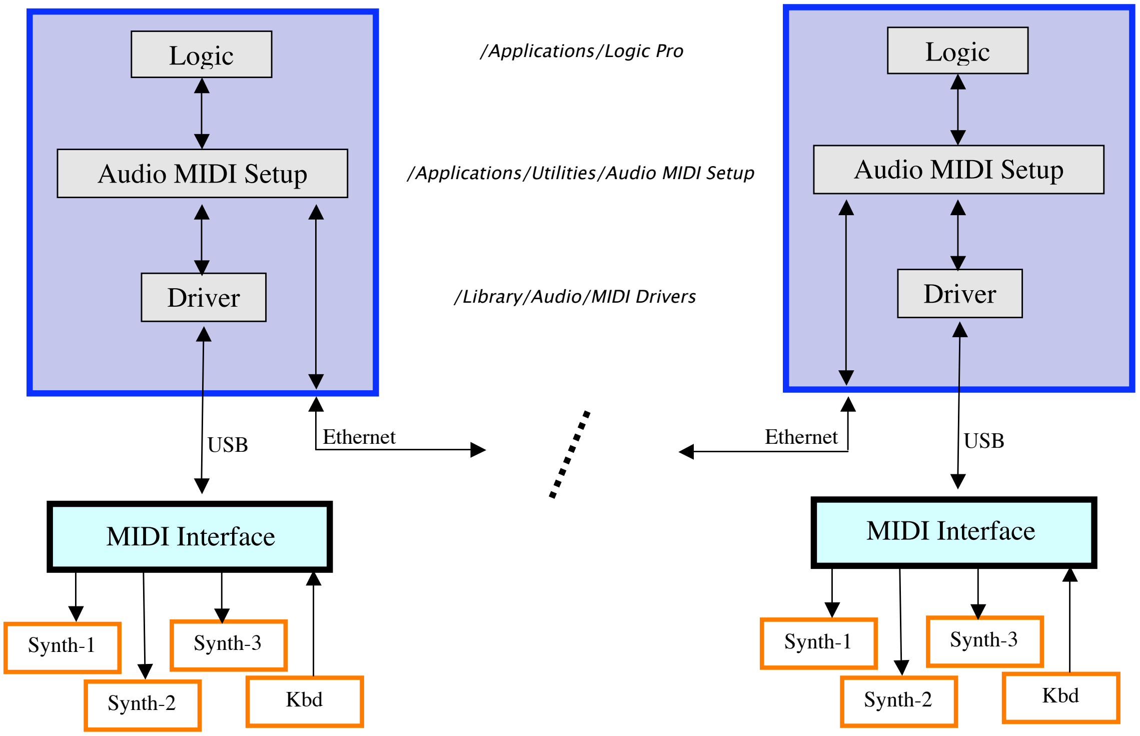
There is only one central utility application named "Audio MIDI Setup" which handles the MIDI configuration for your computer. (The utility contains also the setup for its sibling "CoreAudio".)

- Any hardware that sends/receives MIDI messages to and from the computer needs a driver (a little piece of software), placed into the "*Library/Audio/MIDI Drivers"* directory during the installation. The driver makes the hardware visible in the AMS. The hardware could be a regular MIDI interface connected via USB, a MIDI keyboard connected directly to the USB port, or any other device, with a MIDI interface connects via PCI card or FireWire cable.
- Any software that handles MIDI messages (i.e. a sequencer) gets their information from AMS. MIDI software however use different parts of information from the AMS. Logic and ProTools lookup most of the AMS configuration and make them available right in their program where Cubase kind of ignores most of the user setup and looks up only the driver. This is a poor implementation, because you have to reconfigure your MIDI setup in your application again even if it is already available in the AMS. (This reflects a little bit the old "PC mindset" of Cubase where each application exists in their own environment and isn't able to share recourses. Compare the management of plug-ins under Windows vs OSX and you know what I mean)

User Setup:
AMS provides a graphical user interface that enables you to create the MIDI signal flow based on how you hooked up your MIDI devices. If your sequencer software is smart enough, then it will provide that setting in its own environment. You can even store different configuration in AMS for different workflows or hardware setups.

Below is an example of the MIDI signal flow and the centralized role of the AMS:

- All the MIDI synthesizer are connected vie MIDI cable to the MIDI out of your MIDI interface
- The MIDI Keyboard is connected via a MIDI cable to the MIDI in of your MIDI interface
- The MIDI Interface is connected via USB to the computer
- AMS detects the connected MIDI Interface through its driver
- Logic retrieves the MIDI configuration from AMS on startup and makes the devices available in the Environment
- AMS can route MIDI messages from the "outside" (external MIDI devices) and from the "inside" (from Logic) through its Ethernet port. This makes sending and receiving MIDI messages over the network to/from other networked MIDI system possible.



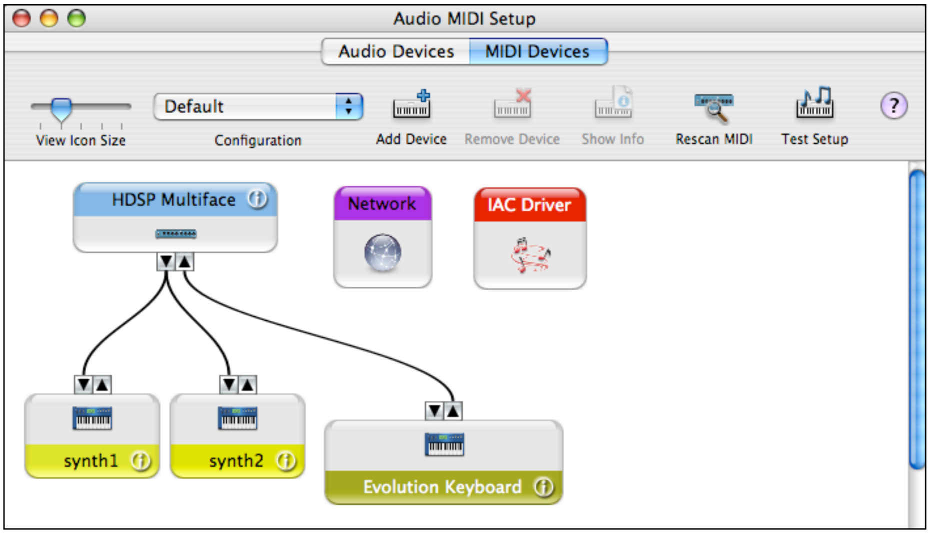
Audio MIDI Setup Utility

The Audio MIDI Setup utility (AMS) provides a very simple user interface with three areas:

1) **Tabs**: selects either the "Audio Devices" or "MIDI Devices" setup window

2) **Toolbar**: icon based commands for managing your configuration

3) **Graphical MIDI Configuration**: lets you configure the routing of MIDI Messages between all the MIDI Devices



What is the purpose of the Graphical MIDI Setup?

Technically you don't need to do anything in there if you just use a straightforward MIDI Sequencer. Your MIDI application will "see" your MIDI interface and all its in and out ports if it has the proper driver installed.
If you know that your Keyboard is connected to "MIDI in 1" then you can select that port for recording your MIDI sequences.
If you know what MIDI out ports are connected to what sound module, then you just route your sequencer tracks accordingly.

You only have to use the Graphical MIDI Configuration:

- If you want to utilize the "*Network*" device or the "*IAC Driver*" device
- If you want to create your MIDI setup and use the following advantages
  - You have a nice visual overview of your whole MIDI setup
  - You can troubleshoot your MIDI setup from inside AMS with its "*Test Setup*" feature
  - Logic looks up the "*User created Devices*" and imports their names and connections direct into the Environment.
  - You can fine tune the settings for each device in its properties window

How to use the Graphical MIDI Configuration:

The GUI works with objects called **Devices**. AMS differentiates between two kinds of Devices: **System created Devices** - **User created Devices**

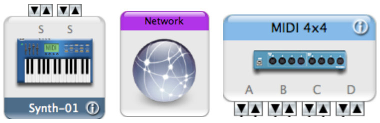
1) System created Devices:

- **MIDI Interface**: AMS automatically creates a device for any connected MIDI Interface with a valid driver. It detects when you connect or disconnect any MIDI interface without the need of a computer restart. If AMS created a Device, which is not longer available (indicated by a lighter shaded color of the Device) then you can remove it.
- **Network:** This device lets you configure all the necessary setting for sending and receiving MIDI over the network (new since OSX 10.4). See below for details
- **IAC Driver:** The Inter-Application Communication driver is a device which allows MIDI applications to transfer information to each other

2) User created Devices

- CoreMIDI detects only the MIDI Interfaces through their drivers but it can't look through your interface and see what MIDI devices you have hooked up to the MIDI in and out connectors of your MIDI interface. You have to manually add your MIDI devices to the AMS and connect them to the MIDI Interface with virtual path chords..

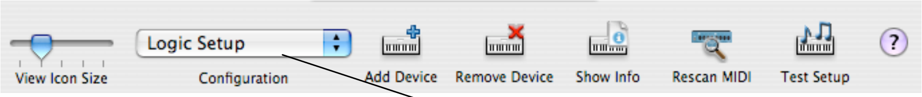
A User created Device has its ports (triangle) at the top and the Device name at the bottom
A System created Device has its ports (if any) at the bottom and the Device name on top

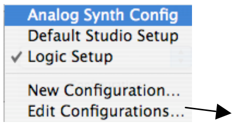


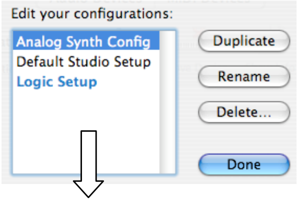
You can manage your MIDI configuration with the following commands:

- **Create Devices**: use the *Add Devices* and *Remove Devices* icons
- **Edit Devices**: dclick on a Device to open the Property Window with all the available settings for that device
- **Connect Devices**: create a virtual patch chord by click dragging between the MIDI out ports and MIDI in ports (little Triangles)

Toolbar:





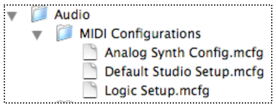


View Icon Size:
changes the size of the Devices

Configuration:
Displays the current MIDI Configuration. Any changes in the GUI will be automatically saved to the current Configuration on disk. There is no "save' command. The popup menu opens a list that displays:

- All the available Configurations (the current one has a check mark)
- *"New Configuration"* command to create a configuration from scratch
- *"Edit Configuration"* command which opens a sheet with:
  - A list with all Configurations (the active configuration in blue font)
  - *Duplicate, Rename, Delete* commands to manage the list.

The configurations are stored as XML files in the User directory "*~/Library/Audio/MIDI Configurations/*



Add Device:
This will add a new user device that you can connect to the MIDI interface

Remove Device:
Removes the selected "user created device". The "Network" and IAC Driver" and connected MIDI interfaces cannot be removed. The only exception is an inactive MIDI interface. This is a MIDI interface Device that was once created but now is not connected, indicated by a lighter shade of its color.

Show Info:
Opens the Property Window of the selected Device. Same as dclick on the Device itself

Rescan MIDI:
You can rescan your computer for any connected MIDI interfaces If i.e. the computer looses connection to a MIDI interface (lighter color) for any reason.

Test Setup:
A click on that button toggles the Test Setup mode off: Test Setup or on: Test Setup the cursor changes to a "eight note" symbol
You can test your MIDI setup in two ways: Activate the button

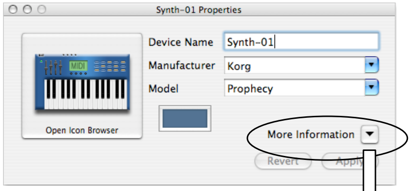
- Clicking on an out connector (triangle) of a MIDI interface will send a MIDI message from that out port to the connected MIDI device. The triangle and the receiving device illuminate. You can also click on the device and it will receive a MIDI message form the MIDI interface out port which it is connected to.
- If you generate a MIDI message on your MIDI Keyboard or MIDI controller (the hardware, not AMS device) and it is connected to a MIDI interface that is visible in AMS, then you will see the MIDI port triangle illuminate to indicate an incoming MIDI signal.

Help:
A click on the question mark will open the Help Viewer for the AMS.

Device Properties Window:

Dclick on a Device or click on the "*Show Info"* in the toolbar will open the Properties window for the selected Device.

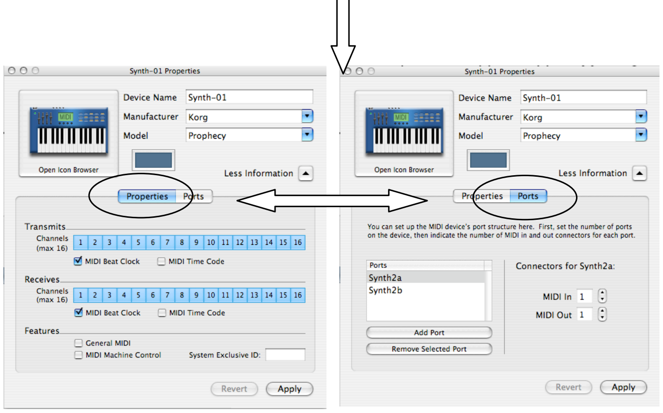
The upper part:
Displays the General Setting
The Triangle Button next to the "More Information" / "Less Information" can expand to reveal the lower part of the Properties Window



Device Name: Synth-01
Manufacturer: Korg
Model: Prophecy

The lower part:
Displays the Detailed Settings, which are divided in a "*Properties"* and *"Ports"* section. Switch between the two section by selection the tab

All Devices have the same windows layout with slightly different settings. The Network Device however has its own Properties windows with completely different settings



Device Name: Synth-01
Manufacturer: Korg
Model: Prophecy

Transmits
Channels (max 16): 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
☑ MIDI Beat Clock ☐ MIDI Time Code

Receives
Channels (max 16): 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
☑ MIDI Beat Clock ☐ MIDI Time Code

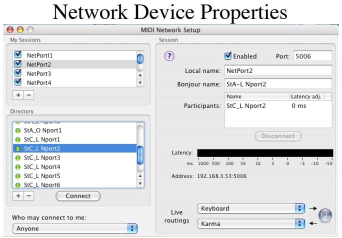
Features
☐ General MIDI
☐ MIDI Machine Control System Exclusive ID:

You can set up the MIDI device's port structure here. First, set the number of ports on the device, then indicate the number of MIDI in and out connectors for each port.

Ports
Synth2a
Synth2b

Connectors for Synth2a:
MIDI In 1
MIDI Out 1

Network Device Properties



My Sessions:
☑ NetPortl1
☑ NetPort2
☑ NetPort3
☑ NetPort4

Directory:
StA_O Nport1
StC_L Nport1
StC_L Nport2
StC_L Nport3
StC_L Nport4
StC_L Nport5
StC_L Nport6

Who may connect to me: Anyone

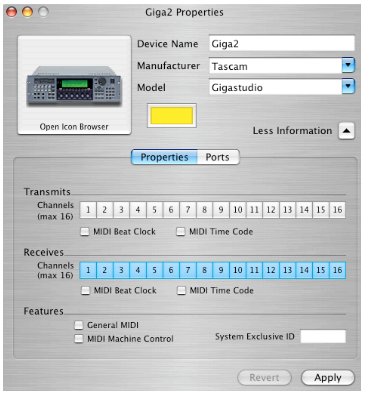
☑ Enabled Port: 5006
Local name: NetPort2
Bonjour name: StA-L Nport2
Participants: Name: StC_L Nport2 Latency adj.: 0 ms
Address: 192.168.3.53:5006
Live routings: Keyboard → Karma ←

"User Created Device" Properties: (you can leave the window open and select different devices (linked windows)

General
- **Device Name:** This name is used for the Object name. Logic uses that name when it creates that MIDI device in the Environment
- **Manufacturer**: select from a list or type it in
- **Model:** select from a list or type it in
- **Icon:** click hold to select from the menu
- **Color:** click to assign a color for the Device

Properties
- **Transmits Channel 1-16, MIDI Beat Clock, MTC**
- **Receives Channel 1-16, MIDI Beat Clock, MTC**: settings will also affect the Object in Logic Environments
- **Features: General MIDI MMC, SysEx ID**

Ports
- **Ports:** add, remove and name the MIDI ports.
- **Connectors:** assign how many MIDI in and MIDI out connectors are available for each port. Each MIDI connector will be displayed as a triangle on the MIDI Device



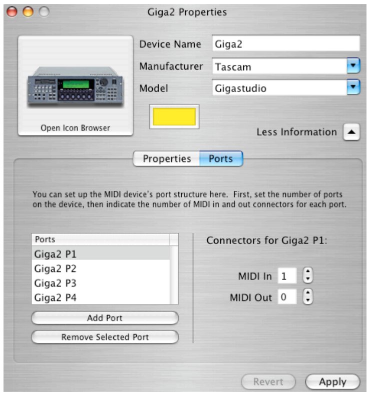
Giga2 Properties

Device Name: Giga2
Manufacturer: Tascam
Model: Gigastudio

Transmits
Channels (max 16): 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
□ MIDI Beat Clock □ MIDI Time Code

Receives
Channels (max 16): 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
□ MIDI Beat Clock □ MIDI Time Code

Features
□ General MIDI
□ MIDI Machine Control
System Exclusive ID



Giga2 Properties

Device Name: Giga2
Manufacturer: Tascam
Model: Gigastudio

You can set up the MIDI device's port structure here. First, set the number of ports on the device, then indicate the number of MIDI in and out connectors for each port.

Ports
Giga2 P1
Giga2 P2
Giga2 P3
Giga2 P4

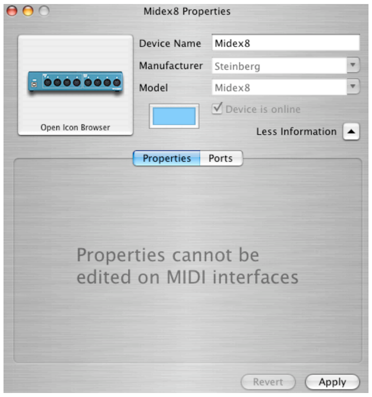
Connectors for Giga2 P1:
MIDI In 1
MIDI Out 0

"MIDI Interface" Properties:

General
- **Device Name:** This name is used for the Object name only in AMS
- **Manufacturer**: read only, retrieved from the driver
- **Model:** read only, retrieved from the driver. Logic uses this name in the Object Parameter Box of the Environment
- **Icon:** click hold to select from the menu
- **Color:** click to assign a color for the Device
- **Device is online:** checked if OS sees the device

Properties
- Cannot be edited

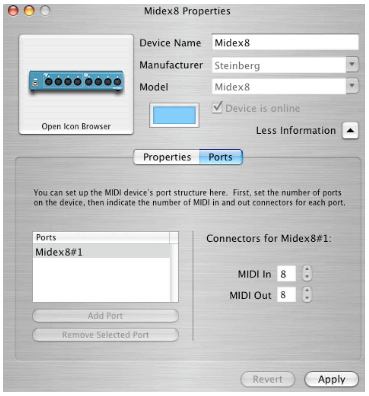
Ports
- **Ports:** read only, retrieved from the driver**.** Logic uses the port name in the Environment
- **Connectors:** read only, retrieved from the driver



Midex8 Properties

Device Name: Midex8
Manufacturer: Steinberg
Model: Midex8
☑ Device is online

Properties cannot be edited on MIDI interfaces



Midex8 Properties

Device Name: Midex8
Manufacturer: Steinberg
Model: Midex8
☑ Device is online

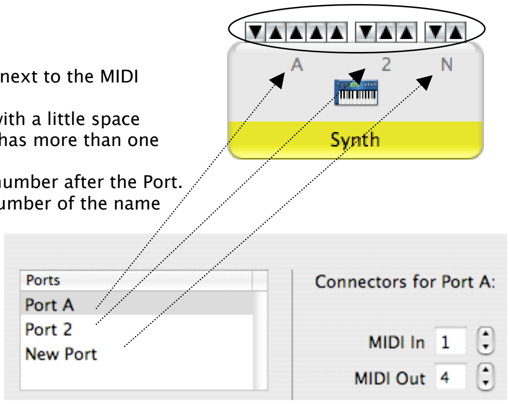
You can set up the MIDI device's port structure here. First, set the number of ports on the device, then indicate the number of MIDI in and out connectors for each port.

Ports
Midex8#1

Connectors for Midex8#1:
MIDI In 8
MIDI Out 8

Ports and Connectors:

AMS follows some rules when you create Ports and Connectors in the Properties Window for a User Created Device:
- A MIDI In connector is displayed by a down triangle
- A MIDI Out connector is displayed by an up triangle
- The MIDI In and Out triangles are grouped together for each port
- To create more connectors for a port, select the port and use the up/down button next to the MIDI number, or type the number in (max number is 8)
- When creating multiple ports, AMS separates the group of triangles for each port with a little space
- AMS displays one number or letter on the Device above or below the triangles if it has more than one port.
- If the name of the port starts with "*Port*", then the AMS displays the next letter or number after the Port.
- If the name of the port doesn't start with *"Port"*, then AMS uses the first letter or number of the name
    - "Port A" → "A"
    - "Port 1" → "1"
    - "New Port" → "N"

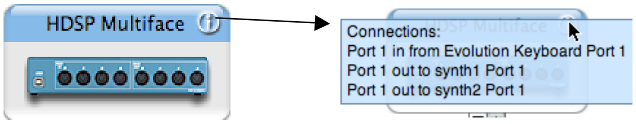


Ports
Port A
Port 2
New Port

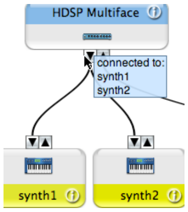
Connectors for Port A:
MIDI In 1
MIDI Out 4

Patch info:

When you connect two devices together, then the "i" icon appears on the right corner of the device. Click on it and a popup menu displays the connections with the name of the ports



Connections:
Port 1 in from Evolution Keyboard Port 1
Port 1 out to synth1 Port 1
Port 1 out to synth2 Port 1



connected to:
synth1
synth2

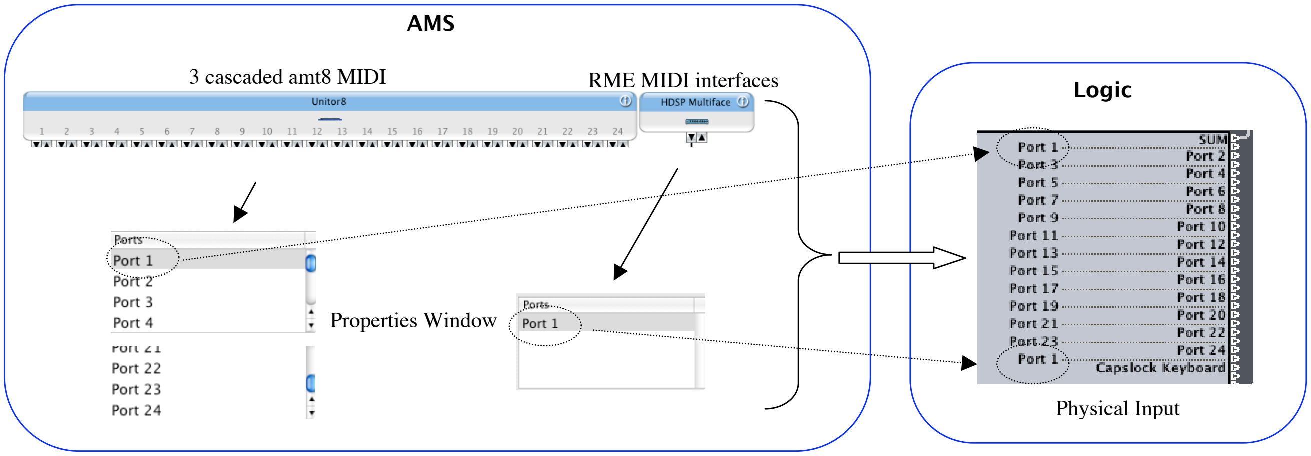
You can also click on a connector and a popup menu displays the name of the connected devices

Note: you can only patch between MIDI Interface Devices and User Created Devices. You can't make a connection between the MIDI Outs of User Created Device to the MIDI In of a User Created Device to simulate a MIDI thru connection!

AMS and Logic

MIDI Interface (MIDI In) → Logic Environment

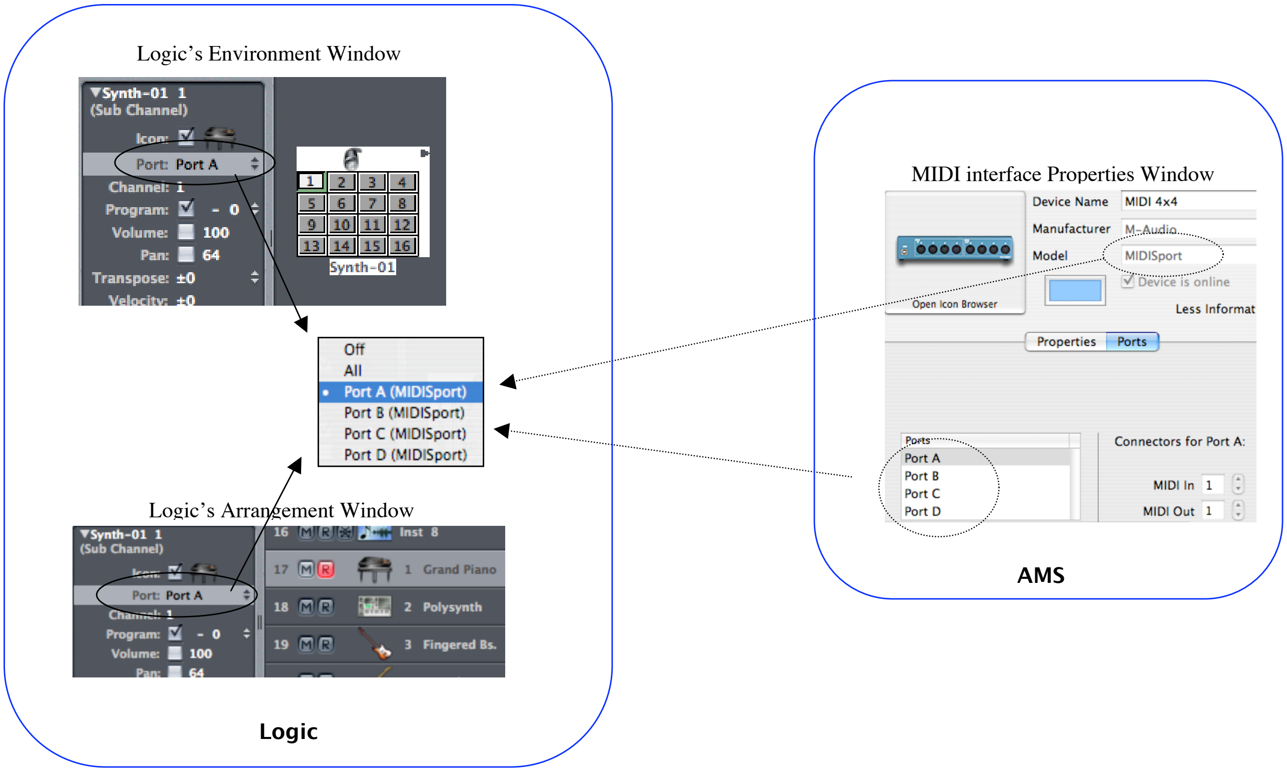
All the Ports of all active MIDI interfaces in AMS will show up in Logic's Environment under the "*Physical Input" object*. These represent the MIDI Ins of your MIDI Interface



The problem is that you cannot rename the Ports of MIDI interfaces in AMS. This can cause some confusion as you see in the example above. If you have two MIDI interfaces and both use the generic "Port 1" label, then in Logic's Environment you see two "Port 1" inputs in the "Physical Inputs" object. Hopefully this will change in a future update, because you can overwrite the labels of the audio inputs of any device then why not the MIDI ports.

Logic → MIDI Interface (MIDI Out)

All the Ports of all active MIDI interfaces in AMS will show up in Logic's "*Object Parameter Box"* also known as "*Instrument Parameter Box"* as the parameter *"Port"*. These represent the MIDI Outs of your MIDI Interface



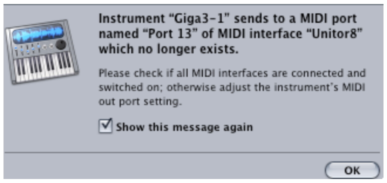
Port: Port A
Channel: 1
Program: ☑ - 0
Volume: 100
Pan: 64
Transpose: ±0
Velocity: ±0

Device Name MIDI 4x4
Manufacturer M-Audio
Model MIDISport

Connectors for Port A:
MIDI In 1
MIDI Out 1

The popup menu lists all available ports of all MIDI interfaces. The name structure is *"Ports Name" ("Model Name")*. The Model Name in parentheses is only visible in the popup menu and not in the Parameter Box itself. Again, it would be nice if you could custom label the Ports of a MIDI Interface.

Missing MIDI Interfaces



Instrument "Giga3-1" sends to a MIDI port named "Port 13" of MIDI interface "Unitor8" which no longer exists.

Please check if all MIDI interfaces are connected and switched on; otherwise adjust the instrument's MIDI out port setting.

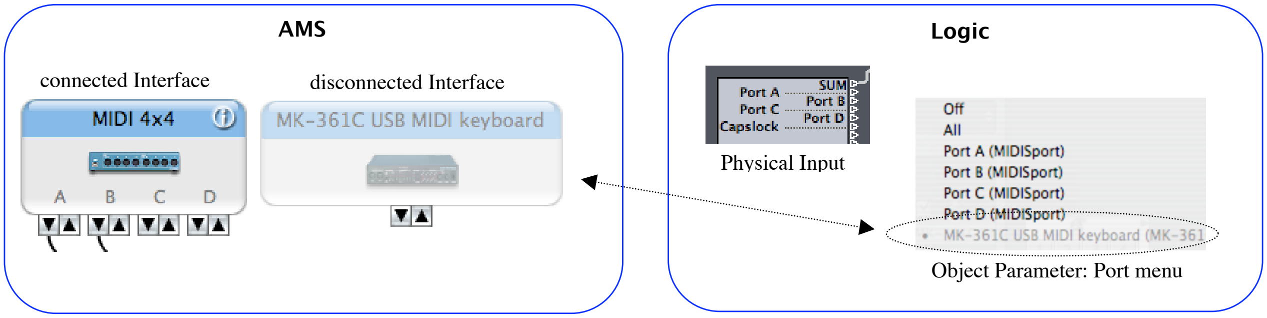
☑ Show this message again

OK

If you loose any MIDI interface connection while Logic is running, CoreMIDI will detect it and Logic displays a warning message with the information of the lost MIDI ports and the Instrument it was connected to.

Once a MIDI interface was recognized from AMS, it stays in the Configuration display. When you disconnect it or it looses its connection, the device turns to a lighter color. The proper color returns when AMS re-establishes the connection. You can remove the inactive device if you no longer need it. You can only remove in-active MIDI interface devices, not active ones!

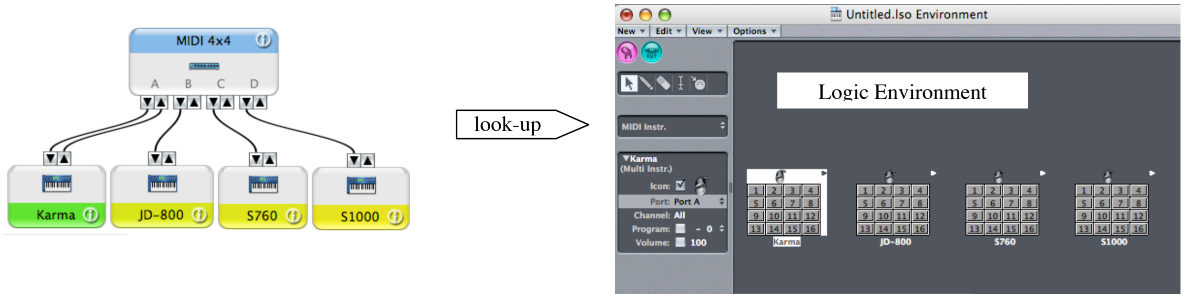
The "Port" popup menu in Logic's Object Parameter Box will list an unavailable MIDI Interface in a grayed out font.
The "Physical Input" object in Logic's Environment doesn't list inactive MIDI interfaces, only the active ones. If you loose a connection then the Port will not be visible anymore.



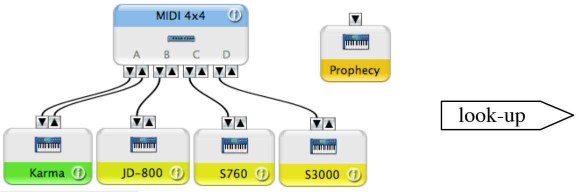
Custom Created Devices Look-up

The Environment Setting will be saved as part of a Logic Song and therefore all the Instruments you created will be part of the Song. When you open any song (including the Autoload Song), Logic will ignore the Custom Created Devices in AMS and it looks up only the available Ports of the connected MIDI Interfaces. If any Port is not available when you open a Song, Logic will display an error message with the information of the missing port and the Instrument it was connected to (see above "Missing MIDI Interfaces"). The name of the Port is still visible in the Object Parameter Port menu, but with a light grey font.

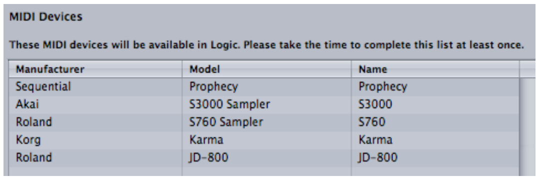
There is one exception where Logic retrieves the Custom Created Devices from AMS. When Logic opens its "Default Song". This is the Song that opens when there is no Autoload Song defined and you launch Logic or select "New" from the File menu without a template. In that case Logic looks up the Custom Created Devices in the AMS configuration and creates an Instrument for each device and assigns it to the port it is connected via a patch chord in AMS. Logic also uses the Device Name to name the Instrument in the Environment.



Actually there is one more exception, when you run the Logic "*Setup Assistant"* either the first time after you install Logic or by using the "*Logic > Preferences… > Start Logic Setup Assistant"* command. The Setup Assistant brings up at one point the "*MIDI Devices"* window, which is a look-up from the AMS. It lists all the Custom Created Devices from AMS, but creates Instrument Objects only for the MIDI Devices that are connected to a MIDI Interface. In the example below the Device "Prophecy" is listed in the "MIDI Devices" Window but it will not show up as a MIDI instrument in the Environment later.



Logic Setup Assistant



MIDI Devices

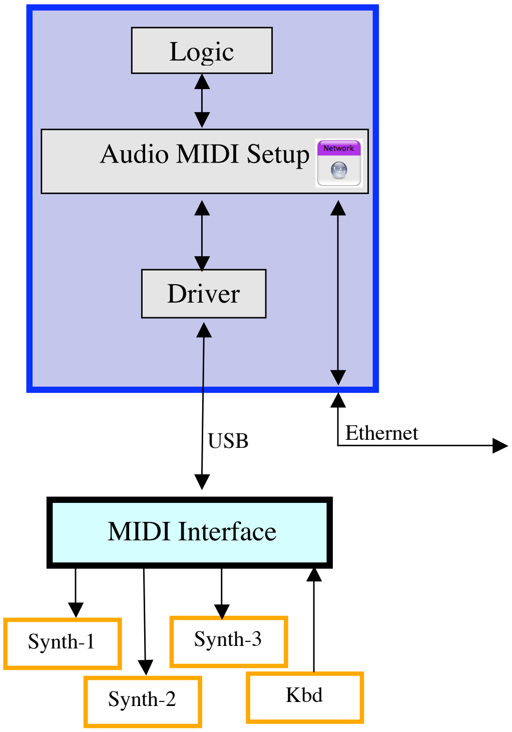
These MIDI devices will be available in Logic. Please take the time to complete this list at least once.

Manufacturer	Model	Name
Sequential	Prophecy	Prophecy
Akai	S3000 Sampler	S3000
Roland	S760 Sampler	S760
Korg	Karma	Karma
Roland	JD-800	JD-800

MIDI Network Setup

This Network Device is available since OS X 10.4 (Tiger)

It enables you to route MIDI messages not only through the connected MIDI interfaces on your local computer but also through the Ethernet port and send the MIDI messages to another computer on your local network or even on the internet.



All the discussion so far were dealing with the standard MIDI environment with MIDI Interfaces, MIDI Devices, Ports, channels etc. You should have no problem with MIDI configurations with the basic understanding of those concepts .

Entering the relam of MIDI Networking however requires a little background information in networking: The “MIDI environment” with MIDI messages, channels, ports, in and out connectors has to conform to the rules of the “Network environment” in order to make it happen.

All the required setting for that edevaur will be made in the “*Network”* device of the AMS.

You will notice that there is a lot of non-MIDI terminology going on in the Network MIDI setting, so lets make a detour into Network land

Network Background

IP Address
Every Computer on a network (local or internet) has to have a unique address. This is the “IP Address” like “234.112.3.4”
The analogy would be the postal address of a building. You send a package with a valid address and the postal service will deliver it to the correct recepient.

Network Port
If the mail carrier finds the right building to deliver the package, he could be confronted with a problem. The building is hundreds of floors high with thousands of appartments. It looks like the package contains a CD, maybe a demo CD and maybe in the building is a record label which could be the recepient of that package, but which floor. If the address would contain a departement number in addition to the street address, then a correct delivery of that package would be guaranteed.
This is where the “*Port Number”* comes in. Every package sent over a network does contain a port number attached to the IP Address. Port Numbers are assigned to specifc “*Services”*. If you type an IP Address in your browser to retrieve a specific website, it might look like http://132.24.123.2. What you don’t see is that your browser actually adds the port number to the IP Address which then looks like this: http://132.24.123.2:80 with the added port number 80 separated by a colon. Port Number 80 stands for Webservice. The Browser knows that you requesting a webpage and adds the necessary port number for you. Now if the request arrives at the computer with the IP Address 132.24.123.2, it know, based on the port number 80, to direct the request to the web server and not the ftp server or any other Service on that computer. The Webserver gets the request and send the correct website back to you.
For more information about ports and a list of port numbers and their assigned services look at
http://docs.info.apple.com/article.html?artnum=106439

Bonjour
The is an Apple technology formerly known as Rendezvous. It is based on the Zero-config network technology. Without getting into much detail, it provides a functionality that lets network services on a LAN detect themself without any configuration:
As I mentioned above, all network services have an assigned network port (web=80, afp=548, iChat=5297, iTunes=3689). Bonjour enabled services lets you assign an easy to remember ”Bonjour Name” to the more cryptic “*IP address:Port*” numbers. That Bonjour Name is visible to all computers on the local network without any configuration. Example:
ITunes has a build in network share feature that uses Bonjour. If you go to iTunes preferences and turn on the music sharing, the only thing you have to do is to enter a Bonjour Name (it’s called “Shared Name”) and your iTunes music cues will be visible to everybody on the LAN under your Bonjour Name. You don’t have to configure any network stuff, you don’t have to know the network stuff of th other computers on your network, nor do the other computer on your network have to know your network stuff in order to play your cues: Zero Config.
The same works for sharing pictures in iPhoto, sharing printers, seeing all available buddies for iChat or just accessing all availble computers through file sharing.

Network Device – MIDI Interface Device

With the Network Device you actually create a “Virtual Network MIDI Interface”

- Instead of having a limited number of MIDI ports with a hardware MIDI interface, you can create as many network MIDI ports as you need
- The network MIDI ports will show up in Logic in the same place as the regular MIDI ports of your connected MIDI interfaces
- You can route MIDI messages from/to your network MIDI ports to any Device in your AMS setting.

Step 1) Create your Network MIDI Port (Session)

In the left upper corner of the MIDI Network Setup window you will find the “My Sessions” Box.
I think this is a bad name choice from Apple’s side. They should have named it what it is: “Network MIDI Ports” or something similar which makes the MIDI functionality clear.
You can add or remove any Network MIDI Port (Session) with the plus and minus sign. The checkbox next to the session name lets you activate/deactivate any Network MIDI Port

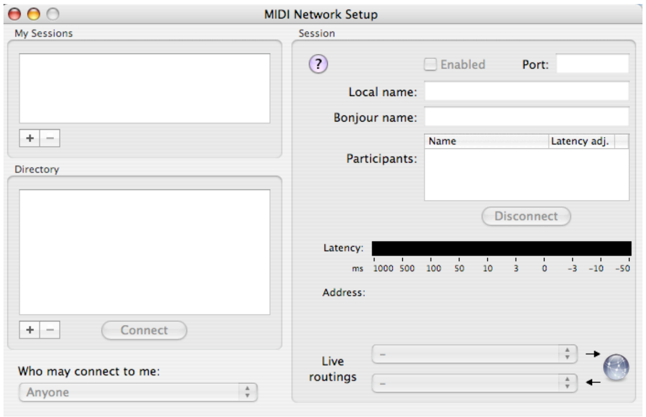
Step 2) Configure your Network MIDI Port (Session)

The right side of the MIDI Network Setup window has all the settings necessary to configure your MIDI Network Port (Session):

Enable
This enables the Network MIDI Port (Session) and has the same functionality as the checkbox next to the Session name box

Port
A network port number will be automatically assigned when you create a Network MIDI Port (Session). The first port number will be 5004, then 5006, and so on. Maybe this is the reason why Apple didn’t choose the term “Network MIDI Port”, to avoid confusion with the “Network Port”. I still think it would be better to have both “Port” names in the configuration to make it clear what port you are dealing with.
The Network Port number will be automatically added to the “Address” field below.

Local Name:
When you add a new Network MIDI Port (Session) with the plus sign below the “My Session” box, AMS will enter the default name “Session 1”, Session 2” etc. You can overwrite that name either directly in the “My Session” list by double clicking on the name or you change it in the “Local name” field.
The local name is important, because this is the name which will be displayed in Logic.
“Local name” = “Network MIDI Port” name = “Session” name



Network Property Window

Bonjour name
This is the important part to understand Apple’s MIDI Networking:
MIDI Networking is a Network Service like iTunes, iPhoto or FileSharing. What does that mean?

Compare the two Network Services:

iTunes	MIDI Network
Activating iTunes music sharing enables that Network Service • Your computer has already a specific IP Address • iTunes Music Sharing has a dedicated Port: 3689 • You enter a Bonjour name which represents the IP Address + Port numb • Everybody on your network with iTunes can see your Bonjour name for iTunes and can access your music • iTunes provides one Network service	Creating a Network MIDI Port (Session) enables the Network Service • Your computer has already a specific IP Address • AMS creates a dedicated Port: 5004, 5006, ... (one port for each Network MIDI Port • You enter a Bonjour name which represents the IP Address + Port number • Everybody on your network with AMS can see your Bonjour Name for that specific Network MIDI Port and can send/receive MIDI messages through that port • Provides a separate network service for each Network MIDI port (Session)

Please note that each MIDI Network Port represents a separate Network Service defined by its own Bonjour Name which represents the IP Address+Port Number.

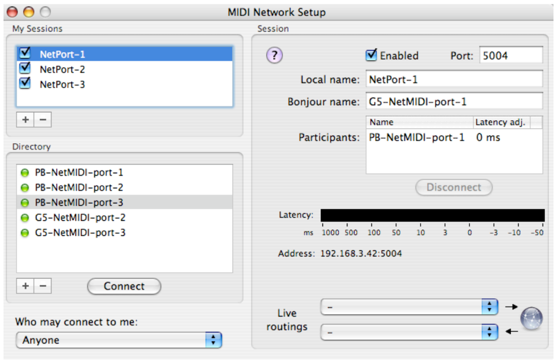
Step 3) Establish Connections

Directory
The list displays all Bonjour Names (Network Services) it discovers automatically on the local network that represent a “Network MIDI Port”. That includes also the own Network MIDI Ports, except the one that is selected in the Session box above. The plus and minus button lets you connect/disconnect Network MIDI Ports outside the LAN (Internet).
The “Connect” button adds the selected Bonjour name to the “*Participants”* list

Participants
This list displays all the Bonjour names (MIDI Network Ports) that can exchange MIDI messages with the selected MIDI Network Port over the network.
The “Disconnect” button removes the selected Bonjour Name (MIDI Network Port) from the Participants list.

Example

Computer A: Powerbook



MIDI Network Setup

My Sessions

☑ NetPort–1
☑ NetPort–2
☑ NetPort–3

Session

☑ Enabled   Port: 5004

Local name: NetPort–1

Bonjour name: G5–NetMIDI–port–1

Participants:

Name	Latency adj.
PB–NetMIDI–port–1	0 ms

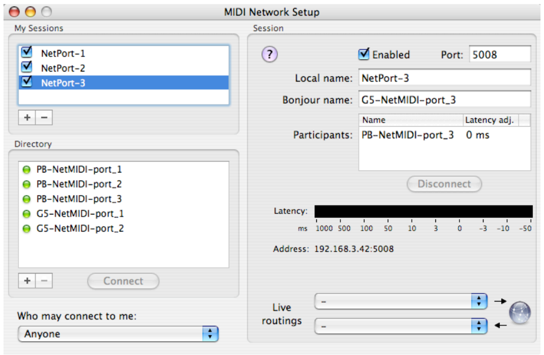
Directory

- PB–NetMIDI–port–1
- PB–NetMIDI–port–2
- PB–NetMIDI–port–3
- G5–NetMIDI–port–2
- G5–NetMIDI–port–3

Address: 192.168.3.42:5004

Who may connect to me: Anyone

Computer B: G5



MIDI Network Setup

My Sessions

☑ NetPort–1
☑ NetPort–2
☑ NetPort–3

Session

☑ Enabled   Port: 5008

Local name: NetPort–3

Bonjour name: G5–NetMIDI–port_3

Participants:

Name	Latency adj.
PB–NetMIDI–port_3	0 ms

Directory

- PB–NetMIDI–port_1
- PB–NetMIDI–port_2
- PB–NetMIDI–port_3
- G5–NetMIDI–port_1
- G5–NetMIDI–port_2

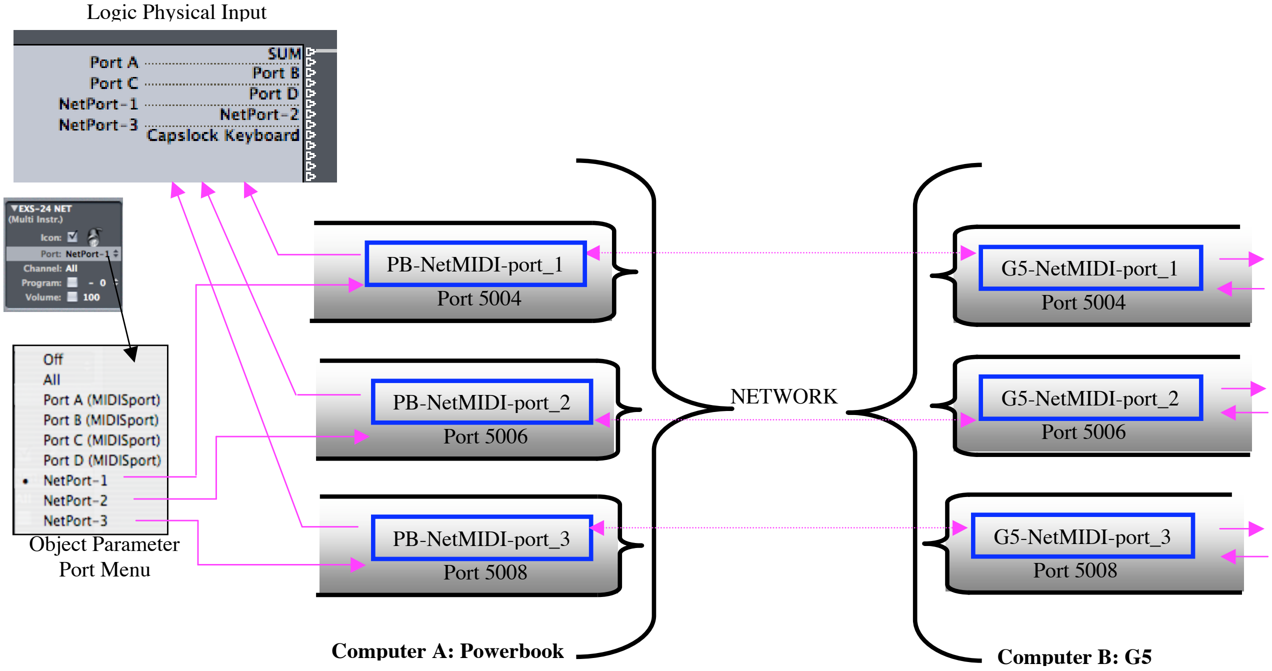
Address: 192.168.3.42:5008

Who may connect to me: Anyone

- Two networked computer: Powerbook and a G5
- Both machines have 3 Network MIDI Ports (Sessions) defined.
- You can choose the same Local names for the Network MIDI Port on both computers, because the name is only used inside the local Logic application.
- Choose a Bonjour name for each of the Network MIDI Ports. This name will be distributed on the network and should be easy to identify.
    - Assign a MIDI network Port available on the Network (Directory) to each local MIDI Network Port (Session):
    - Add the PB MIDI Network Port 1 (PB–NetMIDI–port_1) to the G5 Network MIDI Port 1 Participants list
    - Add the PB MIDI Network Port 2 (PB–NetMIDI–port_2) to the G5 Network MIDI Port 2 Participants list
    - Add the PB MIDI Network Port 3 (PB–NetMIDI–port_3) to the G5 Network MIDI Port 3 Participants list
- You have to make the connection for each port only on one machine. The other machine will recognize that it was added by another computer and will automatically add that computer to its on Participants list for the specific MIDI Network Port

Local name	IP Address	Network	Address	Bonjour Name	Linked to
(Network MIDI Port name)	of your computer	Port			(Bonjour Name)
Computer A: Powerbook					
NetPort–1	198.168.3.52	5004	198.168.3.52:5004	PB–NetMIDI–port_1	G5–NetMIDI–port_1
NetPort–2	198.168.3.52	5006	198.168.3.52:5006	PB–NetMIDI–port_2	G5–NetMIDI–port_2
NetPort–3	198.168.3.52	5008	198.168.3.52:5008	PB–NetMIDI–port_3	G5–NetMIDI–port_3
Computer B: G5					
NetPort–1	198.168.3.42	5004	198.168.3.42:5004	G5–NetMIDI–port_1	PB–NetMIDI–port_1
NetPort–2	198.168.3.42	5006	198.168.3.42:5006	G5–NetMIDI–port_2	PB–NetMIDI–port_2
NetPort–3	198.168.3.42	5008	198.168.3.42:5008	G5–NetMIDI–port_3	PB–NetMIDI–port_3

As you can see below, the MIDI Network Ports look like real MIDI Ports of a MIDI Interface. Like with the real MIDI Ports, you have to know what is connected in order to make the proper MIDI routing for the receiving and sending MIDI messages. At least you can name your Network MIDI Ports to describe their ports depending on you setup configuration.



By creating a Network MIDI Port on Computer A and let it “participate” with a Network MIDI Port of Computer B, you create a virtual MIDI cable through the Network (pink dotted line), connecting both Network MIDI Ports.

Live Routing

Example A
The example above illustrates to possible connection of two separate Logic applications running on two separate computers, sending MIDI back and force over the network through “virtual MIDI cables”. One computer could run Logic as a regular sequencer and the second logic computer could function as a audio instruments playback machine, where you send the MIDI messages from you Logic tracks of Computer A to the (max 128) audio instrumnets of Computer B.

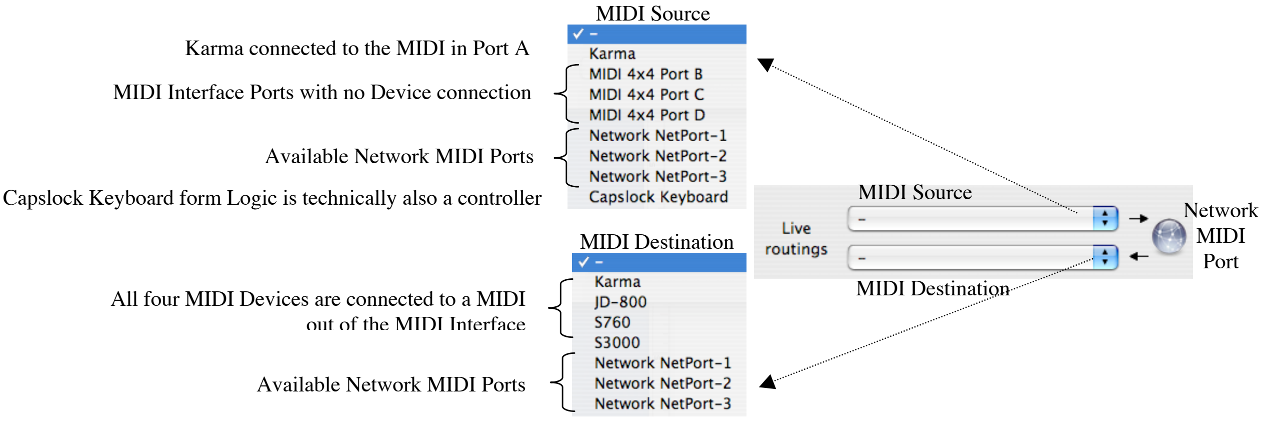
Example B
You could have your G5 hooked up to all the MIDI gear in the studio and when you or a client comes in with his Logic song on his Powerbook, you don’t need to transfer the song and all additional media files to the G5. You just have both computers connected to the network and configure the Network Device on both machines, so you can send the MIDI messages direct from your Logic app on your Powerbook to the G5 and all its connected MIDI gear.

Example C
You can use the Network MIDI Ports also for a network jam session, where all the participating computers contribute their connected MIDI gear to a “virtual pool” of availabel MIDI devices and you can setup AMS which controller plays which sound device.
You accomplish this with a feature called **Live Routing**

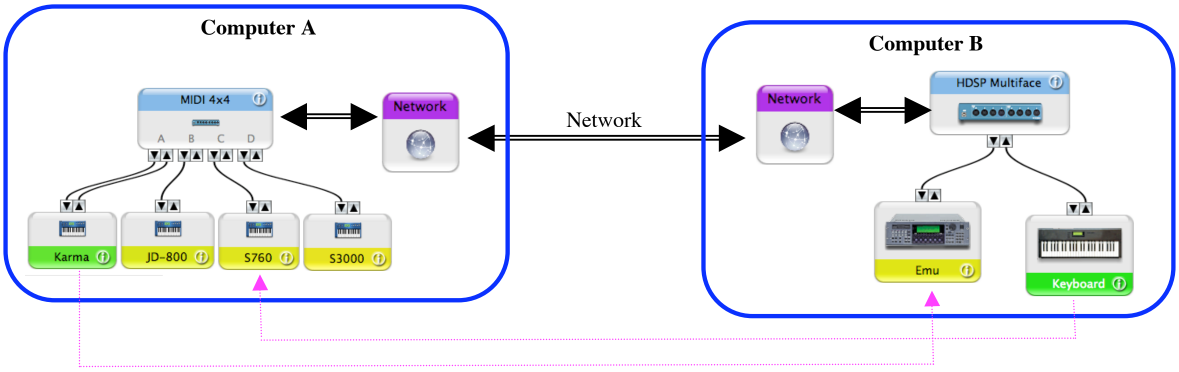
For the Logic app Network MIDI Ports aren’t any different then real MIDI Ports from a connected MIDI interface. They show up in the “Physical Input” object as a MIDI In Port and also appear in the popup menu of the Instrument Object Parameter Box as MIDI Out Port. So you have to use a MIDI application if you want to play any MIDI device over the network.
The “Live Routing” feature however lets you use a Network MIDI Port to connect any MIDI Device in the AMS configuration window of Computer A with any MIDI Device in the AMS configuration window of computer B. i.e Your USB Keyboard on your Powerbook can play the Gigastudio connected to your G5’s MIDI interface.

Here are the rules:

- Each individual Network MIDI Port (Session) has its own Live Routing settings
- Upper Popup menu lists all the available Devices that can send MIDI messages: MIDI Source
    - Any MIDI Device that is connected to a MIDI in connector of your local MIDI Interface
    - If there is no Device connected to the MIDI In connector of the MIDI interface, then the Port Name of the MIDI Interface will be listed
    - All the Network MIDI Ports will also be listed with the prefix “Network”
- Lower Popup menu lists all the available Devices the can receive MIDI messages: MIDI Destination
    - Any MIDI Device that is connected to a MIDI out connector on your local MIDI Interface
    - All the Network MIDI Ports will also be listed with the prefice “Network”



By using the Live routing feature in the illustration below, you can play the Emu on Computer B from your Karma (connected to Computer A) and at the same time play the S760 (on Computer A) from the Keyboard connected to Computer B as if they would be connected through virtual MIDI cables (pinl dotted lines)

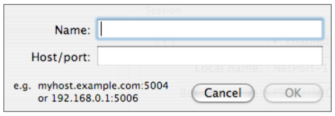


Summary:

My Session:
List of local “Network MIDI Ports” with add and remove button

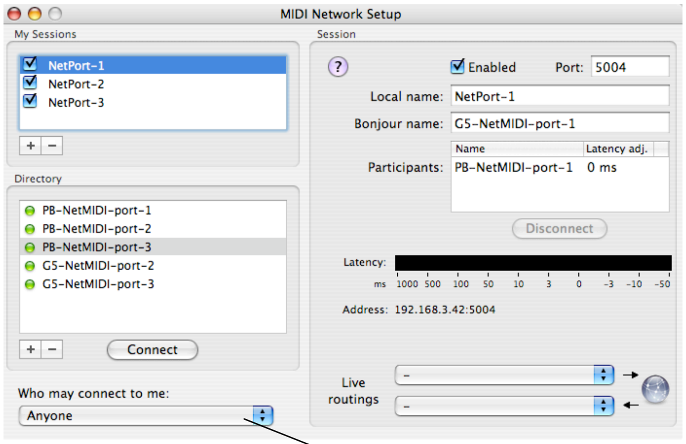
Directory:
List of all available “Network MIDI Port” on the local Network, including local Network MIDI Ports and manually added prom outside the LAN.

Add/Remove Directory
Add “Network MIDI Ports” from computers not on your local network (internet)
Enter a Name an the URL into the window



Name:
Host/port:
e.g. myhost.example.com:5004
or 192.168.0.1:5006

You can only add and remove non-local Network MIDI ports. Bonjour detects Network MIDI Ports on the local network automatically and updates the list



MIDI Network Setup

My Sessions
☑ NetPort-1
☑ NetPort-2
☑ NetPort-3

Directory
PB-NetMIDI-port-1
PB-NetMIDI-port-2
PB-NetMIDI-port-3
G5-NetMIDI-port-2
G5-NetMIDI-port-3

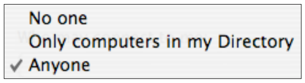
Who may connect to me: Anyone

Session
☑ Enabled Port: 5004
Local name: NetPort-1
Bonjour name: G5-NetMIDI-port-1

Name	Latency adj.
PB-NetMIDI-port-1	0 ms

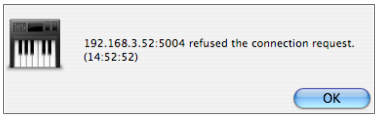
Address: 192.168.3.42:5004

Connect:
Enter the selected Network MIDI Port to the Participants list. This will establish a virtual MIDI connection between the selected local Network MIDI Port and the Network MIDI Port from the Directory list.



Who may connect to me:
Select from three options to decide how to handle incomming Network MIDI Port connection requests:

- **No one:** refuses all incoming Network MIDI requests. The Computer who tries to connect to you will be promted with an error message
- **Only computers in my Directory:** Requests are limited to Computers on the local network
- **Anyone:** Connects to all requests form local and non-local Networks



192.168.3.52:5004 refused the connection request.
(14:52:52)

Network MIDI Port configuration
Displays the parameters of the selected MIDI Network Port in the Mys Session list

- **Enable:** enable/disable checkbox for the selected Network MIDI Port
- **Port:** “Network Port” assigned to the “Network MIDI Port”
- **Local name:** this is the name for the Network MIDI Port how it appears in MIDI applications
- **Bonjour name**: this is the name for the Network MIDI Port how it appears on the network
- **Participants:** list of all MIDI Network Port (their Bonjour name) that communicaate with the selected Network MIDI Port
- **Disconnect:** Remove the selected Participant from the list
- **Latency:** displays the latency of the MIDI message when it travels over the Network
- **Address:** display of the complete network address of the current Network MIDI Port (IP address:Port number)

Live routing:
Establishes connections between MIDI Devices (sending and receiving) availabel in AMS and the selected Network MIDI Port

---

Please Note !!!

Don’t quit AMS after you establish Network MIDI Network connections with other computers. Quitting the AMS app will disonnect all MIDI network connections and when you re-open AMS all the Participants lists are erased. You have to re-connect the Network MIDI connections. I don’t know if this is a bug or if there is a reason for it.

You can save and recall different AMS configurations based on your studio setups. The settings in the “MIDI Network Setup” however stay the same independent of what AMS configuration you are using.