GlobalTalk in QEMU – A Quick Guide

with Extraneous Notes and Observations

by europlus (@europlus@social.europlus.zone on Mastodon) of **WOzFest Labs**, Australia

v1.0.5, 2025-02-25

About

While the easiest way to join **GlobalTalk** as an Apple Internet Router (AIR) node is to use original hardware, the beauty of emulation is the ease of replication.

This document provides information and links to help configure the qemu-system-m68k emulation engine as an Apple Internet Router and AppleTalk client to participate in **GlobalTalk**.

QEMU can only emulate a Quadra 800, and to date, installing AIR in System 7.1 seems easiest/most stable. The files accompanying this Read Me allow you to either setup QEMU from scratch and install all the required components in turn (including base System 7.1), or to use an image with System 7.1 and Apple Internet Router 3.0 pre-installed, just waiting for you to create a new Zone to join **GlobalTalk**.

I highly recommend the latter path. Feel free get everything installed from scratch, and I'll see you back here when you've got a pristine install of all components ready.

Requirements are:

- An emulation host operating system compatible with qemu-system-68k – this set of files has configuration files compatible with host OSes as follows (other versions and OSes may work):
  - macOS Sequoia host
  - Ubuntu 22.04 and 24.04
  - Windows 11
- At least 10GB of disk space to allow for extraction of the images, and perhaps for testing and backing up of those .img files

While I can't *guarantee* support for the included information, files, and applications, I am more than happy to receive DMs in relation to all this to try and help more people join GlobalTalk via emulation.

[1] #MARCHintosh Event Logo concept and design by J. Rivera

Contents

The archive you have downloaded includes:

- **Read Me** – this document.
- **qemu-macOS-GlobalTalk.command** – the command line file to boot from the GlobalTalk HD image in macOS
- **qemu-Ubuntu-GlobalTalk.command** – the command line file to boot from the GlobalTalk HD image in Ubuntu
- **qemu-OCI-GlobalTalk.command** – the command line file to boot from the GlobalTalk HD image in Ubuntu in an Oracle Cloud Infrastructure (OCI) instance
- **qemu-Win11-GlobalTalk.bat** – the command line file to boot from the GlobalTalk HD image in Windows 11
- **GlobalTalk_HD.img** – the disk image with System 7.1 and Apple Internet Router 3.0 pre-installed, mounted as boot disk in the above .command/.bat files
- **Sys7.1-GlobalTalk_install.img** – a bootable disk image with additional installers and apps, mounted as a secondary disk in the above .command/.bat files
- **Q800.ROM** – the ROM file required to boot qemu-system-m68k as a Quadra 800, referenced in the above .command file
- **pram.img** – the emulated Mac's Parameter RAM (PRAM) file referenced in the above .command file – supplied to make your installation slightly easier
- **declrom** – a ROM file for the classicvirtio virtual NuBus card for improved cursor handling and share folder from the host OS
- **share** – an empty share folder which should be available from within System 7.1

How to Use These Files – macOS

1. Download QEMU: go to https://www.emaculation.com/forum/viewtopic.php?t=11728 and download the latest QEMU system available there
2. Create a folder in, say, your **/Applications** folder, called **qemu-system-m68k**.
3. Place the files accompanying this Read Me in the folder created in Step 2.
4. Move the following files from the QEMU downloaded in Step 1 into the folder created in Step 2:
   allow_qemu_to_run-m68k.command
   Libs_arm64
   Libs_x86_64
   qemu-system-m68k
5. Right-click on the folder created in Step 2 and choose **New Terminal at Folder** in the **Services** sub-menu.
6. Type
   `xattr -c allow_qemu_to_run-m68k.command<return>`
7. Then type
   `sudo ./allow_qemu_to_run-m68k.command<return>`
8. Then type
   `sudo ./qemu-macOS-GlobalTalk.command<return>`
   and enter your admin password when prompted, this starts the QEMU system and boots from the GlobalTalk HD image.

How to Use These Files – UTM on macOS

1. Download UTM: go to https://docs.getutm.app/installation/macos/ and download the latest QEMU system available there
2. Download and unzip the GlobalTalk.UTM.zip file from https://drive.google.com/file/d/15NQqslYvav4lomtnG8scRmoNKZJ3ljOt/view?usp=sharing This file in based on a System 7.1 config file from https://github.com/adespoton/utmconfigs
3. Double-click on the GlobalTalk.UTM file.
4. Click Play on the GlobalTalk virtual machine.
   Note: The files in the main GlobalTalk Setup folder are not used in UTM – I mention it here in case you have gone to the other files first, but have already installed, or wish to install UTM for AIR/GlobalTalk use.

How to Use These Files – Ubuntu

1. Instructions on how to install QEMU, create a bridged network connection and use the configuration files mentioned here are at https://blog.europlus.zone/community/marchintosh-wrap-up-globaltalk-news

How to Use These Files – Ubuntu on OCI

1. It is beyond the scope of this document to describe all the steps to create an Ubuntu instance on OCI. You will need:
   a. An Ubuntu instance with at least 10GB free disk space for the GlobalTalk files – more if you are planning on installing Netatalk and hosting files within that system
   b. While an AMD64 shape with a single core and 1GB RAM is adequate, consider a more powerful shape (including an ARM one with more cores and RAM) if this will be a primary AIR node for you.
   c. An entry to allow UDP port 387 in the Security List for the VCN the instance is attached to.
   d. Firewall rule/s to allow port UDP 387 to be accepted by the Ubuntu system.
2. Install QEMU: download and compile QEMU using instructions at https://blog.europlus.zone/community/marchintosh-wrap-up-globaltalk-news
   Note: ***Do not*** follow the instructions beyond the “sudo make install” command, bridged networking as described will not work in an OCI instance.
3. Create a folder at **/home/ubuntu/qemu**
4. Place the files accompanying this Read Me in the folder created in Step 3.
5. In the terminal type:
   `sudo /home/ubuntu/qemu/qemu-OCI-GlobalTalk.command<return>`
   Note: sudo is needed because the UDP port 387 is forwarded as is to the guest OS, and such permissions are needed for ports below 1024 to be forwarded in this way.

How to Use These Files – Windows 11

1. Install QEMU: follow the install instructions at https://tinkerdifferent.com/threads/globaltalk-global-appletalk-network-for-marchintosh-2024-and-beyond.3392/page-4#post-35819

   Note: The .bat file in this package already incorporates those instructions’ changes for Windows compatibility

Next steps

Once the OS has successfully booted, follow the instructions in the *Apple Internet Router Administrator's Guide—Addendum* (available at https://drive.google.com/drive/folders/1ZCZok5wRWGB3eLsRRyJrf3mRyHMmPrpV?usp=sharing) to configure your Apple Internet Router.

> **Note:** When configuring the guest system in an OCI instance, use these MacTCP settings to correlate with the defaults in the .command file.
>
> 
>
>
> Further information on the basis of these settings can be found at https://www.qemu.org/docs/master/system/devices/net.html#using-the-user-mode-network-stack

System Software 7 Pro* – with PowerTalk!

While it ***is*** possible to update to System 7.1.1, AIR 3.0.1 and Network Software 1.5.1. I've left the main release at the same versions Paul Rickard originally described, but I have successfully updated these components in my own setup.

System 7.1.1 brings AppleScript and PowerTalk (which includes AppleMail [serverless e-mail/messaging]). The installers are in the GlobalTalk Installers folder on the install image, just follow the additional numbered installers in order.

*aka System 7.1.1

Known Issues

If you meddle with your setup (most especially System-related installs or updates), your system may report that the EtherTalk port is not available for AIR when starting up. If this happens, try opening the Network control panel and choosing EtherTalk again. If *that* fails, Shut Down the QEMU emulated machine, then start it again with the sudo command above. You should be able to open the Network control panel, and choose EtherTalk successfully. Once EtherTalk is successfully selected, you can open the AIR Router Management app and choose "Start Router..." from its Control menu.

Assorted Miscellany

- If you use Disk Jockey https://diskjockey.onegeekarmy.eu/ to open the images supplied here, you can extract the individual partition with the data as an image, which is mountable (and even bootable) in mini vMac II (and possibly other emulators).
- Classic Mac (HFS) allocation block size is a function of disk size. O...K...(I hear you say) WTAF does that mean? Basically, HFS volumes can only have 65,536 allocation blocks – the larger the disk, the larger the allocation block size (think file size quantum).

My boot image is 2GB, so each allocation block is 2GB÷65,536=32kB. A 1-byte file will be allocated a 32kB block! At 32,769 (32k+1) bytes, a second 32kB block will be allocated to that file, using 64kB on the 2GB volume. This is less than optimal.

The wastage is worse the more small files there are on a volume as a percentage of the total number of files. The solution is basically multiple volumes of smaller size. Historically, that's meant partitioning (which can still be done in QEMU with the 2GB image – split it into 2 x 1GB partitions, and the allocation block size halves to 16kB on each partition, for example). In emulation, you can just mount multiple disk images of smaller sizes.

No matter what you do, understanding this "one weird trick" of Mac disk sizes is useful. I wrote an article about this in the September 1996 issue of *Australian MacUser* – I'll try and scan it (or find a link to it) to refence here.

Credits

Special thanks to the following people, without whom the WOzFest HQ **GlobalTalk** zone and this guide would not be possible:

QEMU Maintainers https://www.qemu.org – for creating an emulator engine which is compatible with both System 7.1 and bridged Ethernet networking

Macintosh Garden, Macintosh Repository, Emaculation, and **other archive maintainers** – without having the required installers and information files available, none of this would be possible

The Retrocomputing Community on Mastodon – I've deleted my Twitter and Facebook accounts, and steer clear almost 100% of Slack and Discord. While this might seem like I'm being *more* insular, I have in fact found my community of peers and involvement have both expanded and been richer (not monetarily – *any* involvement in retrocomputing does not make you monetarily richer) – love you guys and gals and others! <3

No doubt countless other giants on whose shoulders I stand.