



## Jamulus Introduction and Pre-Requisites

Produced by Nás Na Rí Singers and published by Sing Ireland.

Hi all,

My name is Stephen O'Reilly, a Tenor with Nás Na Rí Singers. I'd like to introduce you to a software package called Jamulus which has enabled me to connect with other choral singers during Covid 19 lockdown. I met a few like minded singers on facebook and we managed to pull together an international choir. We called it the **Jamulus Chamber(s) Choir** as we were all singing from our respective "chambers". A recording of the collaboration of a few choirs in June is available on youtube <https://youtu.be/m811fZiPyaQ?t=562>

The software package at first sight seems quite technical, but when you get into it it is actually quite basic. It sets out to do one job and to do it well, and in my opinion it succeeds hands down.

In my experience since the beginning of April, I have found that there are a few items that need to be seriously considered to make the software work well and in reality if any of these items are avoided or bypassed, the results you will experience will be less than adequate and will affect your experience and that of others with whom you are trying to make music. In this document I intend to set out the main requirements below and a short explanation of why they are important. I hope you don't run away from the technical nature of them but trust me please. Read the main bullet points at the very least.

### Jamulus Prerequisites

- Jamulus software (latest version)
- Audio Interface
- Quality microphone
- A Good Internet
- Windows 10 / Mac / Unix
- Wired Headphones
- Ethernet cable
- High power mode, not power saving on PC/laptop.
- Sing using a full Score
- Microphone stand

- Jamulus software (latest version) currently version 3.6.0 as of 25/10/2020
  - Jamulus is free software with no onward running costs.
  - Jamulus is an active open source development, the developer Volker Fischer is constantly introducing new features and improvements.
  - Only the very latest versions of software will cater for large choirs and groups. Version 3.5.13(beta) server has been tested to 100 virtual clients without degradation of sound quality.
  
- Audio Interface with native low latency ASIO driver -
  - ASIO driver reduces the latency. Latency is the time taken for sound to be processed and returned back to you by the system.
  - Direct control of input gain - this is your input volume which needs to be balanced with the input of other singers in the server.
  - Professional quality low noise preamp.
  - Using Jamulus without a reasonable audio interface will introduce low quality sound and increase low level noise in the virtual room. It also increases the complexity as the ASIO4ALL driver is very tricky to set up and keep working.
  
- Quality microphone
  - A poor mic or one designed for speech levels will not perform well, and bring poor quality sound or noise into the virtual room. Inexpensive mic's are not designed to handle the higher volumes of singing or musical instruments, only for speech.
  - A professional quality microphone is required, these days you can get a very good quality microphone without spending a lot of money.
  - USB microphones will add to latency as they do not usually have ASIO compatible drivers. They also rarely have a suitable output for headphones. On Windows, the output from Jamulus needs to be the same device as the input, using an interface provides this.
  
- A Good Internet with a short Round Trip Time (RTT) to the server.
  - A RTT of less than 20ms is ideal but not always possible. As this number increases the echo you perceive will lengthen and it will get more distracting. I have managed to sing in time with singers across Europe and with practice it is possible to stay in time with much longer delays over 100ms.
  - We need a continuous stream of narrow bandwidth data for Jamulus, a small amount of data to be transferred in the shortest possible time and tests like **speedtest.net** are not suitable. They are targeted at testing bandwidth which is the amount of data you can transfer in a period of time and is measured in Mb/s. A wide bandwidth connection will allow a large quantity of data in a short space of time and that data may be in large bursts with gaps that cause pops and cracks.
  - Testing RTT can be done with Jamulus or ping.
  
- A good internet with low Jitter.
  - Jitter is defined as variations in the trip time of a stream of data, audio data in this case, which means that portions of that data stream arrive with bursts and gaps and sometimes in a different sequence than which it was sent. Combining this back into a smooth stream is done by the jitter buffer.
  - Jitter can be caused by busy or overloaded routers, route balancing and redirected packets along the route.

- Jamulus will cope with a small amount of jitter which is to be expected on the internet, but higher levels of jitter will degrade the audio stream and it will sound warbled, watery, with clicks and pops.
- A test for a clean low jitter route can be done with **ping**, a command line utility available in Mac Windows and Unix. You can ping your chosen server or a well known server on the internet to see what the variation in reply time is. See testing examples at the end.
- Windows 10 / Mac / Unix is required.
  - Jamulus does not work on mobile devices (phones, tablets or iPads) Jamulus hasn't been written to allow use on mobile devices as they rely on wireless communication topology, which uses larger buffers.
- Wired Headphones are essential.
  - Using loudspeakers will allow sound from the system to be picked up by your microphone and be fed back into the Jamulus creating a feedback loop.
  - We need to do everything we can to shorten the audio path via the internet. Bluetooth headphones add another level of latency and buffering which we need to eliminate. They also may not connect to your audio interface.
- Ethernet cable connected from your PC to the router.
  - Wi-Fi uses buffers that are far too big to allow for a short trip time. It does increase bandwidth. However that doesn't help us here and actually causes problems as the data arrives in larger bursts with bigger gaps. We need a continuous stream for Jamulus.
  - A Cat5 ethernet cable is needed as it does not use large buffers and significantly shortens the trip time.
  - Don't use powerline Ethernet extenders as they add more delay due to encoding and decoding of the data.
- High power mode, not power saving on PC/laptop.
  - Avoid running other applications on the computer. They will take valuable processing time away from Jamulus.
  - Running your pc on mains 220v power and not on battery will most likely allow you to use high power mode. You can also create a power profile to use high power.
  - Jamulus needs to process data at a high data rate to ensure smooth audio using a high precision timer. If your pc is running in low power mode, it will put Jamulus in low priority mode and not respond fast enough to the timer causing more jitter.
- Sing using a full Score (not just a single line for parts)
  - Without a conductor it is best to follow a full score to see your own and other musician's entries, rests and empty bars. It is essential to count to keep with the beat. Careful counting helps to counter any small dropout of sound.
  - Using zoom as a social aid is possible but you cannot use audio or follow a visual meter as it will be behind the audio in time.
- Using a microphone stand is essential.
  - Both hands are free for music.
  - Holding a mic in your hand creates noise everyone can hear, with a group of singers this noise can add up.

## **Important links**

Jamulus software :-- <http://llcon.sourceforge.net/>

Jamulus help at :-- <https://github.com/corrados/jamulus/wiki>

The best driver we have found for the Behringer UMC22 and UM2 is at <http://recordingstudio9.com/about/download/> please make sure to read the readme.

## **Nás na Rí Singers, shopping list.**

Interface : Behringer UMC22 :-- [https://www.thomann.de/ie/behringer\\_u\\_phoria\\_umc22.htm](https://www.thomann.de/ie/behringer_u_phoria_umc22.htm) cost €41 (also available from Amazon)

Microphone : Behringer XM8500 :--- [https://www.thomann.de/ie/behringer\\_xm8500.htm](https://www.thomann.de/ie/behringer_xm8500.htm) cost €16.50 + mic cable €5 approx

Mic Stand : [https://www.thomann.de/ie/millenium\\_ms2005\\_mic\\_stand.htm](https://www.thomann.de/ie/millenium_ms2005_mic_stand.htm) €11.90

Mic + Stand + Lead : [https://www.thomann.de/ie/behringer\\_xm8500\\_bundle.htm](https://www.thomann.de/ie/behringer_xm8500_bundle.htm) cost €33

Studio Headphones : [https://www.thomann.de/ie/superlux\\_hd681.htm](https://www.thomann.de/ie/superlux_hd681.htm) cost €19.60

Savings may be made by ordering audio equipment in bulk to avoid an individual charge of €15 per delivery.

Contact us via [nasnarisingers@gmail.com](mailto:nasnarisingers@gmail.com)

## **Servers**

Since I have been using Jamulus with a few choirs it has made sense for me to set up a chain of private servers with my own private central server. Depending on demand it may be possible to link your servers to this chain for choral use provided the users keep in contact with me to avoid unexpected server outages etc.

The server application is bundled with Jamulus so you could run a server on your own pc. However you would need to set up port forwarding or register with a central server if you need others to connect to it. Also consider your upload bandwidth and how many singers you have as you will need about 1Mb/s per person.

Setting up a server in the cloud is not too difficult, the basic server can manage about 15-20 singers and there are instructions to do this on the Jamulus website. A larger choir will need a more powerful multi core server at the moment. We have successfully tested a 4vCPU 32GB machine up to 70 virtual users.

## Internet quality testing tools.

The best tool available for testing round trip times for Jamulus, is actually Jamulus itself. It uses an uncommon transmission protocol (UDP) and includes test information. At the bottom right side of the settings window there are values for “Overall delay” and “Ping Time”. Ideally these should be under 40ms for delay and a much lower number for ping time. The LEDs also give an indication of quality.

The overall delay on the bottom right corner indicates the time taken for audio to be processed and sent through the complete system including a calculation for your audio equipment. You should expect to see a stable number with a small variation of 1 or 2 ms. The larger the variation in this number, the poorer your audio quality will be. This variation is an indication of jitter. Ping time is just the network component of the trip. The larger these numbers are, the longer the echo you will hear. A ping time of 6 or 7ms is nearly as good as it gets with a good broadband connection.

The second best testing method is ICMP ping which can be viewed/determined from a computer. Opening a command prompt (Windows) or terminal (mac / Linux) and typing in, **ping 8.8.8.8** and enter, will send a echo request to one of Google's dns servers. You can replace 8.8.8.8 with any other destination server. I have shown a few examples below.

I have used 8.8.8.8 below as it's Google's DNS server and is available almost everywhere. The second example is the Central server belonging to the developer of Jamulus and is in Germany. I have highlighted the portions of the result that are important. The first ping result shows no difference in trip time to 8.8.8.8 which is excellent (your mileage may vary) The second example shows a small variation of 1ms which is very good for a server several hundreds of kilometers away.

### Example 1

```
C:\Users\administrator>ping 8.8.8.8
```

Pinging 8.8.8.8 with 32 bytes of data:

```
Reply from 8.8.8.8: bytes=32 time=5ms TTL=117  
Reply from 8.8.8.8: bytes=32 time=5ms TTL=117  
Reply from 8.8.8.8: bytes=32 time=5ms TTL=117  
Reply from 8.8.8.8: bytes=32 time=5ms TTL=117
```

Ping statistics for 8.8.8.8:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 5ms, Maximum = 5ms, Average = 5ms

### Example 2

```
C:\Users\administrator>ping jamulus.fischvolk.de
```

Pinging jamulus.fischvolk.de [62.113.206.102] with 32 bytes of data:

```
Reply from 62.113.206.102: bytes=32 time=30ms TTL=53  
Reply from 62.113.206.102: bytes=32 time=30ms TTL=53  
Reply from 62.113.206.102: bytes=32 time=29ms TTL=53  
Reply from 62.113.206.102: bytes=32 time=30ms TTL=53
```

Ping statistics for 62.113.206.102:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 29ms, Maximum = 30ms, Average = 29ms

*Stephen O'Reilly 2020-11-15 (live doc)*