**This Nancy + MongoDB install was performed successfully on 12th-14th of April.**

**1. Install OS**

Due to the need for also running Mono, I installed Soft-float Debian “wheezy” (2012-08-08-wheezy-armel.zip) to SD using instructions on:

<http://elinux.org/RPi_Easy_SD_Card_Setup>

I used the "Linux + ImageWriter" way of writing the image to SD card.

**2. Update OS to latest:**

sudo apt-get update

sudo apt-get upgrade

After upgrade I have the following:

pi@raspberrypi ~ $ uname -a

Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l GNU/Linux

**3. Configure OS to optimize performance etc:**

run the command:

sudo /usr/bin/raspi-config:

**settings to set:**

memory-split 16MB

overclock: modest 800 MHz

boot to GUI: no

upgrade raspi-config

reboot

**Verify you have the full 512 MB of RAM available:**

after reboot run the command

top

and verify you have almost 500 MB of RAM (on the top left part of page)

**4. Install Mono**

a) sudo apt-get update

b) sudo apt-get install mono-complete (note: takes a \*long\* time, 25+ min)

**5. Nancy**

**I. Compile Nancy:**

a) mkdir /home/pi/code, and cd /home/pi/code

b) sudo apt-get install git ruby rake

c) sudo gem install albacore

d) git clone <https://github.com/NancyFx/Nancy.git>

e) cd Nancy

f) rake mono (takes long time, optionally only run rake compilemono, but I did not do this)

**II. Create Nancy app file:**

a) copy code from <http://techny.tumblr.com/post/36438641616/compile-nancy-applications-on-a-raspberry-pi>

b) write file using nano main.cs and the paste + exit + save

**III. Compile and run Nancy app:**

a) dmcs -r:/home/pi/code/Nancy/src/Nancy/bin/MonoRelease/Nancy.dll -r:/home/pi/code/Nancy/src/Nancy.Hosting.Self/bin/MonoRelease/Nancy.Hosting.Self.dll main.cs

b) mono main.exe

c) invoke service at [http://localhost:8282](http://localhost:8282/)

Note: I copied the two required Nancy assemblies Nancy.dll and Nancy.Hosting.Self.dll into the same directory as my main.exe file so they were placed beside eachother. Must find a way to ensure the will be added to path/GAC or something...

**6. MongoDB**

**I. Install dependencies**

sudo apt-get install build-essential libboost-filesystem-dev libboost-program-options-dev libboost-system-dev libboost-thread-dev scons libboost-all-dev python-pymongo git

(this takes a while...)

**II. Update packages:**

sudo apt-get update

**III. Clone mongopi from Github:**

cd ~/code

git clone git://[github.com/RickP/mongopi.git](http://github.com/RickP/mongopi.git)

**IV. Build mongo:**

cd mongopi

scons

(this takes at least 5 hours and requires that you have completed the steps in section 2+3 to ensure that you have the full 500 MB RAM available...)

**V. Install mongo:**

sudo scons --prefix=/opt/mongo install

(this also takes hours...)

VI. **Add mongo binaries to PATH:**

nano ~/.bashrc

Insert the following line somewhere suitable:

export PATH=$PATH:/opt/mongo/bin/

**VII. Create data directory for mongo and set owner to current user (instead of root):**

sudo mkdir -p /data/db

sudo chown $USER /data/db

**VIII. Start mongo:**

(open a new shell first to ensure the PATH has been correctly set according to .bashrc file)

mongod

**IX. Optional: Clean up space:**

cd <mongopi-dir>

scons -c

**Nice to know:**

**Backup of Raspberry Pi SD card image:**

1) df -h (find SD card device)

2) sudo dd if=/dev/sdd | gzip > /data/raspberrypi\_backup\_wheezy\_soft.img.gz

(in the above command sdd is the device corresponding to my SD card)

**Restore backup onto SD card:**

gzip -dc /data/raspberrypi\_backup\_wheezy\_soft.img.gz | dd of=/dev/sdd

**Useful ressources:**

<http://hubcitylabs.org/unlocking-your-new-raspberry-pis-512mb-of-memory/>

<http://mongopi.wordpress.com/2012/11/25/installation/>

<http://andyfelong.com/2013/02/raspberry-pi-meets-mongodb/>

<http://stackoverflow.com/questions/7948789/mongodb-mongod-complains-that-there-is-no-data-db-folder>

<http://sourcecodebean.com/archives/mongodb-c-and-mono/1408>