Getting started

Create your directories

Open a terminal (ctrl+shift+t).

Create your project directory:

```
cd /path/to/where/you/want/your/study/to/be/
mkdir MyNewStudyDir
cd MyNewStudyDir
```

Create the rest of the directory tree for this project:

```
cd /path/to/your/study/directory/
mkdir Data
mkdir Processing
mkdir Tools
```

And now create template directory

```
cd /path/to/your/study/directory/Tools
mkdir template
```

Copy the template file you want to use as well as its corresponding segmentation and the corresponding probability maps into that template directory (ie: for adult rats, you can copy the files that are in /rodent/atlas/NIRAL/Rat_P72/*date*-current/ for the template file and its segmentation and the files that are in /tools/atlas/rodent/atlas/NIRAL/Rat_P72 /*date*-probability_maps/ for the probability maps.

```
cp /rodent/atlas/NIRAL/Rat_P72/*date*-current/* /path/to/your/study/directory/Tools/template/
cp /rodent/atlas/NIRAL/Rat_P72/*date*-probability_maps/* /path/to/your/study/directory/Tools/template/
```

Copy your data

Copy your data into:

```
/path/to/your/study/directory/Data
```

Each case should have its own folder

Convert the files from /path/to/your/study/directory/Data/{case} to /path/to/your/study/directory/Processing/{case} /1-Converted into an ITK readable format.

Copy MakeAtlas scripts

From here, copy over the MakeAtlas scripts. This will avoid any problems if the developers change the scripts while you are processing your data :

```
cp /rodent/bin_linux64/MakeAtlasScript3/ /path/to/your/study/directory/Tools/MakeAtlasScripts
```

Then create a directory to build your config files :

```
mkdir /path/to/your/study/directory/Tools/MakeAtlasScripts-Config
cd MakeAtlasScripts-Config
```

This is the structure you should have :

■ Study

This is the structure you should have :

- Study
    - Data
        - case1
        - case2
        - case3...
    - Processing
    - Tools
        - MakeAtlasScripts
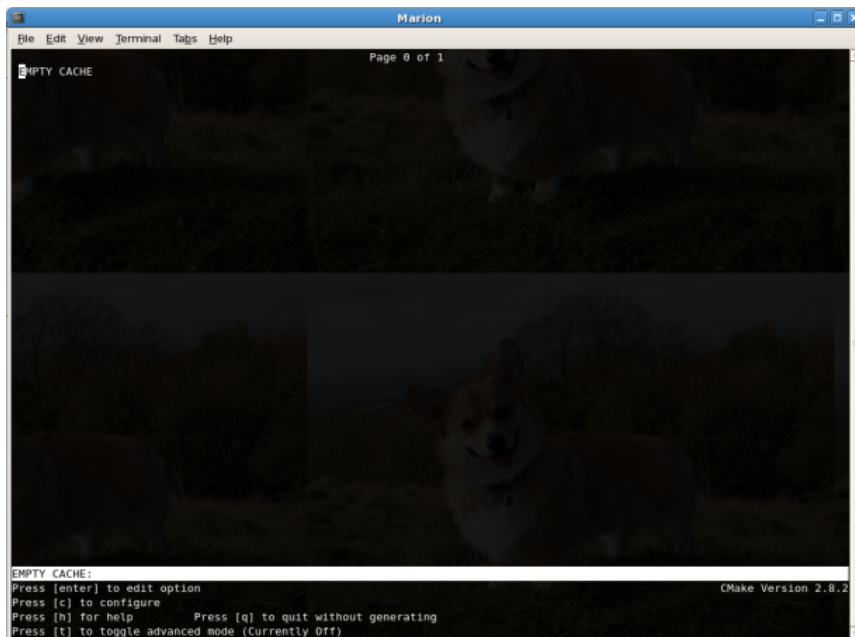   - MakeAtlasScripts-Config

Setting config files with the ccmake
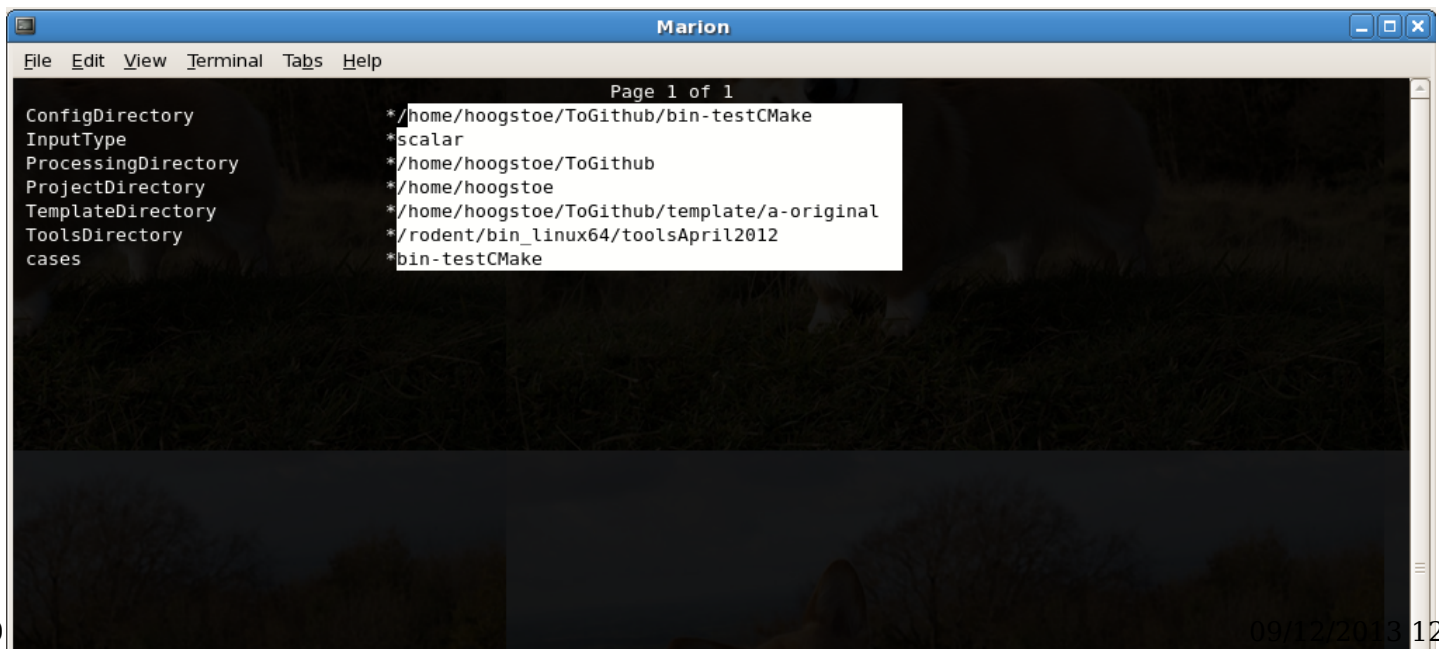
Using the ccmake

Still from your MakeAtlasScripts-config directory, call the CMakeFiles :

```
ccmake ../MakeAtlasScripts
```

Your terminal will completely change. You'll have this :

Press c to configure and you'll have this :

To set what you want to set, choose your line with the arrows, press enter and then write the good path, type or cases names. Please note that you can't use delete on the first letter. Pressing enter will save what you typed and stop the writing mode and you'll be able to choose another line. If you have any doubt about what each parameter is, you have an indication at the bottom of the terminal. The names are as clear as can be, but if it is not enough, this is what each parameter should be :

- ConfigDirectory : the path to your build_MakeAtlasScripts directory
- InputType : scalar, DTI or DWI.
- ToolsDirectory : the directory with the version of the tools being used by rat lab. You shouldn't have to change it
- Cases : names of the cases directories (see in Data directory)

When you'll be done, press c again.

Two things can happen now :

- You don't get any message : press g and proceed to next step
- You get a warning or error message : Don't panic and see section below :)

In case something is wrong/You got an error or a warning message

You got an error or a warning message

Try to find your message in the list below :

Warnings

- "(a tool) was not in (your ToolsDirectory}. The path was set to the dynamic executable path set as default on your machine"

This is no big deal. Just make sure that the tool won't be updated any time soon. (cf Francois)

- "(a tool) not found. Path not set"

This means your tool doesn't exist on the system nor on the tools directory. Please send a note to Francois to tell him :)

Errors

- "InputType must be either scalar or DWI or DTI"

You mispelled your type parameter! It must be (case sensitive) scalar, DWI or DTI.

So, what should I do?

Pressing e will get you out of the message displayer. If you got warnings, you can go on and press g to generate the config files, and proceed to next step If you got errors, you have to correct it (by setting type again), then press c again. You shouldn't have an error message now, so press g to generate the config files. Then proceed to next step. If you still have errors, you may want to start over again

You want to start it all over again

Because MakeAtlasScripts-Config is a separate directory that contains configured files, you can empty it if anything goes wrong without having to do all the Getting started step again. To erase everything that is on your build directory (use with caution!!!!!!!!!!!!!!):

`rm -rf *`

wrong without having to do all the Getting started step again. To erase everything that is on your build directory (use with caution!!!!!!!!!!!!!):

```
rm -rf *
```

Then, start over again to set the config files again.

Using the config files:

If you type :

```
ls
```

This is what you'll find in your build_MakeAtlasScripts :

```
2-RegistrationConfig.bms          4.1-AverageWithHFields.bms      6.1-BRAINSDemonWarpConfig.bms    CMakeCache.txt         GlobalConfig.bms
3-BrainMaskComputation.bms        4-AverageCreationConfig.bms     6-WarpConfig.bms                 CMakeFiles/            Makefile
3.1-BrainMaskComputation-ANTS.bms 5-AverageSegmentationConfig.bms 7-StatsConfig.bms                cmake_install.cmake
```

You can customize each *Config.bms file if necessary, but everything should have been configured properly with ccmake already. You can open each one of them in a text editor to verify that the options and paths are set properly.. For example, open 3-BrainMaskComputationConfig.bms with that commandline :

```
gedit 3-BrainMaskComputation.bms &
```

In that tutorial we'll take the example of processing only three cases named 0001, 0002 and 0003

2-Registration

The config file

Just run the script:

```
BatchMake_1.3 -e 2-RegistrationConfig.bms
```

Wait to see the messages that are gonna be displayed. If you set the config file badly you should have error messages pretty quickly. If that is the case, the script won't stop by itself, so hit CTRL+C. You can try to edit manually the config file.

Registration implements pretty heavy commandlines. You should see something like that :

```
Status: Start /rodent/bin_linux64/toolsApril2012/DWIProcessPipeline   --inputType scalar -t /biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/template/a-or
 run: '/rodent/bin_linux64/toolsApril2012/DWIProcessPipeline' '' '' '--inputType' 'scalar' '-t' '/biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/template
```

And even worse. As long as you don't see Error everywhere, don't panick.

Check the results

The files were scalars

Go check into Processing directory, and in each case directory. Each case should have :

- a registration matrix (e.g 0001_reg.mat)
- a registration image (e.g 0001_reg.nrrd)
- 2-RegistrationConfig.bms
- DTIPipeline.bms

The files were DTIs

Go check into Processing directory, and in each case directory. Each case should have :

- a registration matrix (e.g 0001_reg.mat)
- a registration image (e.g 0001_reg.nrrd)
- a FA (e.g 0001_reg_FA.mat)
- a MD (e.g 0001_reg_MD.mat)
- a ColorFA (e.g 0001_reg_ColorFA.mat)
- a RD (e.g 0001_reg_RD.mat)
- a AD (e.g 0001_reg_AD.mat)
- 2-RegistrationConfig.bms
- DTIPipeline.bms

- a AD (e.g 0001_reg_AD.mat)
  - 2-RegistrationConfig.bms
  - DTIPipeline.bms

The files were DWIs

Go check into Processing directory, and in each case directory. Each case should have :

  - a registration matrix (0001_reg.mat)
  - a registration image (0001_reg.nrrd)
  - a b0 (e.g 0001_reg_b0.mat)
  - a IDWI (e.g 0001_reg_IDWI.mat)
  - a FA (e.g 0001_reg_FA.mat)
  - a MD (e.g 0001_reg_MD.mat)
  - a ColorFA (e.g 0001_reg_ColorFA.mat)
  - a RD (e.g 0001_reg_RD.mat)
  - a AD (e.g 0001_reg_AD.mat)
  - 2-RegistrationConfig.bms
  - DTIPipeline.bms

To do in every case

First, look quickly at the registration image through ImageViewer or any tool of your choice. You should see a brain! Then you can use Slicer to look at the others images (in the case of DTIs or DWIs). If the results are good enough for you, you can move to the next step.

Registration went wrong

If you got errors and then the script stopped without your help before even getting to the heavy commandlines, you're reading the good thing. You can have to error messages :

  - Give max 3 additional images : please reset the parameter ADDITIONAL_IMAGES in the 2-RegistrationConfig.bms file and make sure you don't give more than three images.

  - Give max 2 additional images (NN) : please reset the parameter ADDITIONAL_IMAGES_NN in the 2-RegistrationConfig.bms file and make sure you don't give more than two images.

3-Brain mask computation

What is the difference between 3-BrainMaskComputationConfig.bms and 3.1-BrainMaskComputation-ANTSConfig.bms

Both config files have the same purpose: computing a brain mask that is used in the rest of the pipeline. 3-BrainMaskComputationConfig.bms uses "rreg" and "areg" to perform the registration steps. 3.1-BrainMaskComputation-ANTSConfig.bms uses ANTS. By default, use 3-BrainMaskComputationConfig.bms. If you know what you are doing, then you can use 3.1-BrainMaskComputation-ANTSConfig.bms.

The config file

Check and maybe change 3-BrainMaskComputationConfig.bms.

The pipeline

Save and come back to the terminal.

```
BatchMake_1.3 -e 3-BrainMaskComputationConfig.bms
```

Wait to see the messages that are gonna be dispalyed. If you set the config file badly you should have error messages pretty quickly. If that is the case, hit CTRL+C. You can try to edit manually the config file.

Check the results

Each case

Go check into Processing directory, and in each case directory. Each case should have :

  - a mask (e.g 0001_mask_from_seg.nrrd)
  - a xml file (e.g 0001.xml)

- a mask (e.g 0001_mask_from_seg.nrrd)
- a xml file (e.g 0001.xml)

Now what?

Use Slicer to check the masks. Open also the input images (e.g. 0001_dti_f_reg.nrrd) to make sure the masks are correct. If the results are good enough for you, move on to the average creation (step 4)

3.5-Skullstripping

An extra step that is not necessary for the rest of the pipeline has been added. It allows to skull-strip automatically all the images that were created and rigidly registered in the step 2-Registration.

The config file

Check and maybe change 3.5-SkullStrippingConfig.bms.

The pipeline

Save and come back to the terminal.

```
BatchMake_1.3 -e 3.5-SkullStrippingConfig.bms
```

4-Average Creation

What is the difference between 4-AverageCreation and 4.5-AverageWithHFields?

Average Creation is the whole pipeline. Average With H-Fields is an optionnal additional step where you can recompute the Average. 4.5-AverageWithHFields is useful if you think your measurement frames were wrong (and therefore your tensor orientations were wrong) while computing your average. It computes an average using each subject MD/RD/FA/AD instead of computing an average DTI and computing the average MD/RD/FA/AD from this average DTI. So you need to have completed Average Creation first if you want to use it.

What is the difference between 4-AverageCreation and 4.1-AverageCreation-ANTS?

4-AverageCreation uses AtlasWerks to compute the average image as well as the associated h-fields with that average. 4.1-AverageCreation-ANTS uses a script available in the ANTs package to compute the average image as well as the associated displacement fields.

The config file

Average Creation

Check and maybe change the config file.

Average Creation

```
BatchMake_1.3 -e 4-AverageCreationConfig.bms
```

Average With H-Fields

```
BatchMake_1.3 -e 4.1-AverageWithHFields.bms
```

Wait to see the messages that are gonna be dispalyed. If you set the config file badly you should have error messages pretty quickly. If that is the case, hit CTRL+C. You can try to edit manually the config file.

Check the results

Average Creation

Average Creation

The Average

Go check into Processing directory. There should be :

- an Average directory

e.g:

```
hoogstoe@eurydice(194) Scripts/build$ ls /biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/
0001/  0002/  0003/  Average-Marion/ temp/  template/
```

And inside that directory :

- a temp directory
- the AverageTensor, AverageTensor_AD, AverageTensor_ColorFA, AverageTensor_FA, AverageTensor_MD, AverageTensor_RD
- the same files as above, but SkullStripped

e.g :

```
hoogstoe@eurydice(195) Scripts/build$ ls /biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/Average-Marion/
4-AverageCreationConfig.bms  AverageTensor_FA.nrrd          AverageTensor_SS_AD.nrrd      AverageTensor_SS.nrrd
AverageTensor_MD.nrrd        AverageTensor_SS_ColorFA.nrrd  AverageTensor_SS_RD.nrrd      AverageTensor_AD.nrrd
AverageTensor.nrrd           AverageTensor_SS_FA.nrrd       temp/                          AverageTensor_ColorFA.nrrd
AverageTensor_RD.nrrd        AverageTensor_SS_MD.nrrd
```

Each case

Go check each of the case directories. There should be :

- a directory named 4-AverageCreation

and inside that directory :

- an hField
- an inv_hField
- the config file

I got :

```
hoogstoe@eurydice(199) Scripts/build$ ls /biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/0001/4-AverageCreation-Marion/
0001_hFieldToPopAverage.mha  0001_inv_hFieldToPopAverage.mha  4-AverageCreationConfig.bms
```

Average With H-Fields

The Average

In your processing directory, you should find inside the Atlas-yourSuffix directory something like this :

```
hoogstoe@eurydice(398) Scripts/build$ ll /biomed-resimg/rodent/MIDAS/tutorial_tests/Processing/Average-1/
total 19760
-rw-rw-r-- 1 hoogstoe employee    3140 May 23 10:28 4.1-AverageWithHHFields.bms
-rw-rw-r-- 1 hoogstoe employee 5050327 May 23 10:29 AverageImage_dti_f_reg_AD.nii.gz
-rw-rw-r-- 1 hoogstoe employee 5035995 May 23 10:30 AverageImage_dti_f_reg_FA.nii.gz
-rw-rw-r-- 1 hoogstoe employee 5055756 May 23 10:31 AverageImage_dti_f_reg_MD.nii.gz
-rw-rw-r-- 1 hoogstoe employee 5078509 May 23 10:32 AverageImage_dti_f_reg_RD.nii.gz
drwxrwxr-x 2 hoogstoe employee    4096 May  9 10:40 temp/
```

Now what?

Use Slicer to check some average tensors. You can also check if the skull stripped ones look okay. If the results are good enough for you, use the linkResults scripts and proceed to atlas segmentation.

5-Average Segmentation

Config file

Check and maybe change the config file.

The pipeline

The pipeline

Save and come back to the terminal.

```
BatchMake_1.3 -e 5-AverageSegmentationConfig.bms
```

Wait to see the messages that are gonna be dispalyed. If you set the config file badly you should have error messages pretty quickly. If that is the case, hit CTRL+C. You can try to edit manually the config file.

Check the results

The Average

In the Processing/Average directory, you should find :

- segmentation.nrrd

In the Processing/Average/temp directory, you will also find :

- template_ANTS_Warp.nrrd

Now what?

Use Slicer to check the new segmentation. If the results are good enough for you, use the linkResults scripts and proceed to warp.

Use linkResults

Some explanations

The commandline for the script is : pathToScript/linkResults.script 5 pathToConfigFile/ConfigFile

6-Warp

BRAINSDemonWarp : what is that thing?

It's an additional and optional script that allows you to warp the segmentation to every case with the BRAINSDemonWarp tool. You don't need to process Warp before processing this one. Most of the time, just using Warp is good enough.

Config file

Check and maybe change. Save and come back to the terminal.

Warp

```
BatchMake_1.3 -e 6-WarpConfig.bms
```

BRAINSDemonWarp

```
BatchMake_1.3 -e 6.1-BRAINSDemonWarpConfig.bms
```

Wait to see the messages that are gonna be dispalyed. If you set the config file badly you should have error messages pretty quickly. If that is the case, hit CTRL+C. You can try to edit manually the config file.

Check the results

Warp

For each case

In each case subdirectory in Processing, there should now be a "6-Warping" directory. That directory should contain :

- the segmentation file warped to the specific case
- the config file

- the segmentation file warped to the specific case
- the config file

BRAINSDemonWarp

For each case

In each case subdirectory in Processing, there should now be a "6-Warping" directory. That directory should contain :

- the segmentation file warped to the specific case
- the config file

Now what?

Use Slicer to check the new segmentations. If the results are good enough for you, proceed to stats.

7-Stats

Config file

Check and maybe change :

- ##-1-Cases to process : your input cases

- ##-2-Labels map suffixes (can contain multiple files) : the file names of the output of Warping pipeline. In my example, _segmentation.nrrd

The pipeline

Save and come back to the terminal.

```
BatchMake_1.3 -e 7-StatsConfig.bms
```

That pipeline is wonderfully short. :) Wait to see the messages that are gonna be dispalyed. If you set the config file badly you should have error messages pretty quickly. If that is the case, hit CTRL+C.

Check the results

For each case

In each case directory, there should be a subdirectory called 7-Stats. It should contain :

- the text file containing the stats you just computed. e.g 0002_template_atlas_dti_f_reg_stat.txt
- the config file