

Preprocessing and data QC 2

Nipype (5) : Nipype Basic - Workflow

Sungwoo Lee

M.S-Ph.D. student

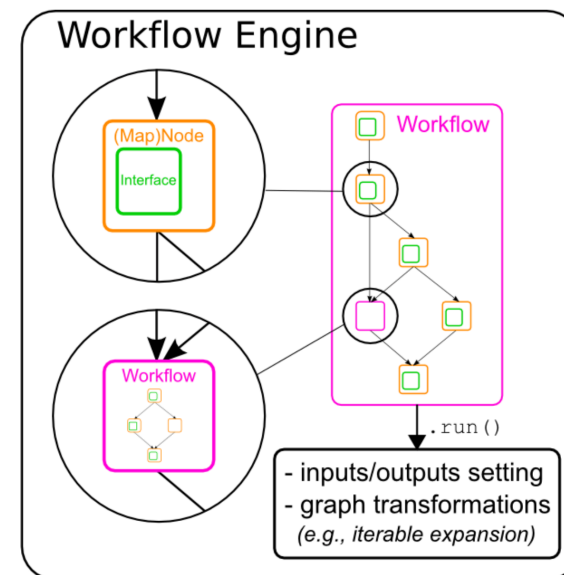


Workflow

Although it would be possible to write analysis scripts using just Nipype [Interfaces](#), and this may provide some advantages over directly making command-line calls, the main benefits of Nipype are the workflows.

A workflow controls the setup and the execution of individual interfaces. Let's assume you want to run multiple interfaces in a specific order, where some have to wait for others to finish while others can be executed in parallel.

The nice thing about a nipype workflow is, that the workflow will take care of input and output of each interface and arrange the execution of each interface in the most efficient way.

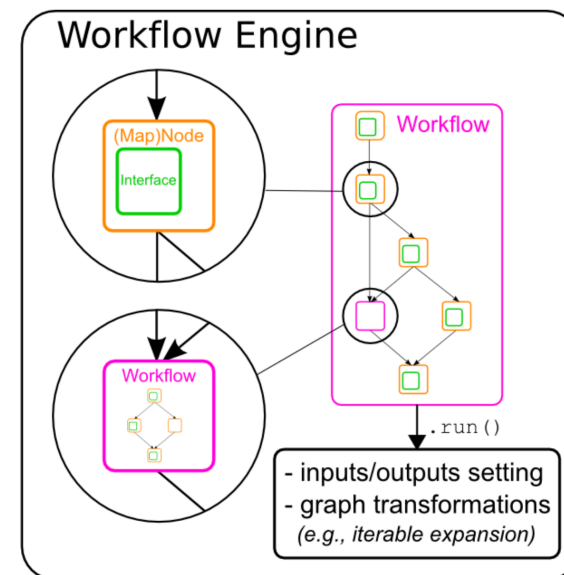


https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

A workflow therefore consists of multiple [Nodes](#), each representing a specific [Interface](#) and directed connection between those nodes. Those connections specify which output of which node should be used as an input for another node.



https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

Interfaces vs. Workflows

Interfaces are the building blocks that solve well-defined tasks. We solve more complex tasks by combining interfaces with workflows:

Interfaces	Workflows
Wrap <i>*unitary*</i> tasks	Wrap <i>*meta*</i> -tasks <ul style="list-style-type: none">implemented with nipype interfaces wrapped inside ``Node`` objectssubworkflows can also be added to a workflow without any wrapping
Keep track of the inputs and outputs, and check their expected types	Do not have inputs/outputs, but expose them from the interfaces wrapped inside
Do not cache results (unless you use [interface caching] (advanced_interfaces_caching.ipynb))	Cache results
Run by a nipype plugin	Run by a nipype plugin

https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

Example

- Skullstrip an image to obtain a mask
- Smooth the original image
- Mask the smoothed image

https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html

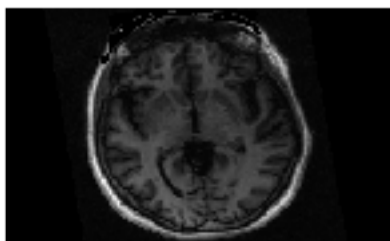


Workflow

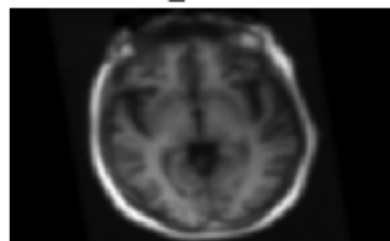
Example : Command Line

```
%%bash
ANAT_NAME=sub-01_ses-test_T1w
ANAT=/data/ds000114/sub-01/ses-test/anat/${ANAT_NAME}
bet ${ANAT} /output/${ANAT_NAME}_brain -m -f 0.3
fslmaths ${ANAT} -s 2 /output/${ANAT_NAME}_smooth
fslmaths /output/${ANAT_NAME}_smooth -mas /output/${ANAT_NAME}_brain_mask /output/${ANAT_NAME}_smooth_mask
```

T1w



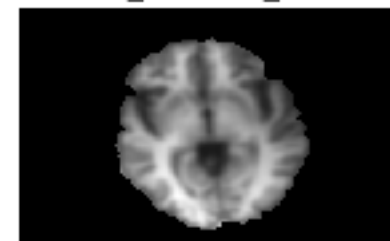
T1w_smooth



T1w_brain_mask



T1w_smooth_mask



https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

Example : Interface

```
from nipype.interfaces import fsl

# Skullstrip process
skullstrip = fsl.BET(
    in_file="/data/ds000114/sub-01/ses-test/anat/sub-01_ses-test_T1w.nii.gz",
    out_file="/output/sub-01_T1w_brain.nii.gz",
    mask=True)
skullstrip.run()

# Smoothing process
smooth = fsl.IsotropicSmooth(
    in_file="/data/ds000114/sub-01/ses-test/anat/sub-01_ses-test_T1w.nii.gz",
    out_file="/output/sub-01_T1w_smooth.nii.gz",
    fwhm=4)
smooth.run()

# Masking process
mask = fsl.ApplyMask(
    in_file="/output/sub-01_T1w_smooth.nii.gz",
    out_file="/output/sub-01_T1w_smooth_mask.nii.gz",
    mask_file="/output/sub-01_T1w_brain_mask.nii.gz")
mask.run()
```

https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

Example : Workflow

```
# Import Node and Workflow object and FSL interface
from nipype import Node, Workflow
from nipype.interfaces import fsl

# For reasons that will later become clear, it's important to
# pass filenames to Nodes as absolute paths
from os.path import abspath
in_file = abspath("/data/ds000114/sub-01/ses-test/anat/sub-01_ses-test_T1w.nii.gz")

# Skullstrip process
skullstrip = Node(fsl.BET(in_file=in_file, mask=True), name="skullstrip")

# Smooth process
smooth = Node(fsl.IsotropicSmooth(in_file=in_file, fwhm=4), name="smooth")

# Mask process
mask = Node(fsl.ApplyMask(), name="mask")
```

https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

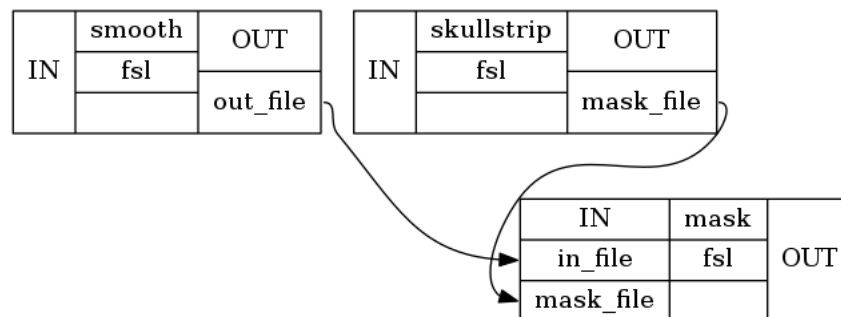
Example : Workflow

```
# Initiation of a workflow
wf = Workflow(name="smoothflow", base_dir="/output/working_dir")

# First the "simple", but more restricted method
wf.connect(skullstrip, "mask_file", mask, "mask_file")

# Now the more complicated method
wf.connect([(smooth, mask, [("out_file", "in_file")])])

wf.write_graph("workflow_graph.dot")
from IPython.display import Image
Image(filename="/output/working_dir/smoothflow/workflow_graph.png")
```



https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow

Example : Workflow

```
# Specify the base directory for the working directory
wf.base_dir = "/output/working_dir"

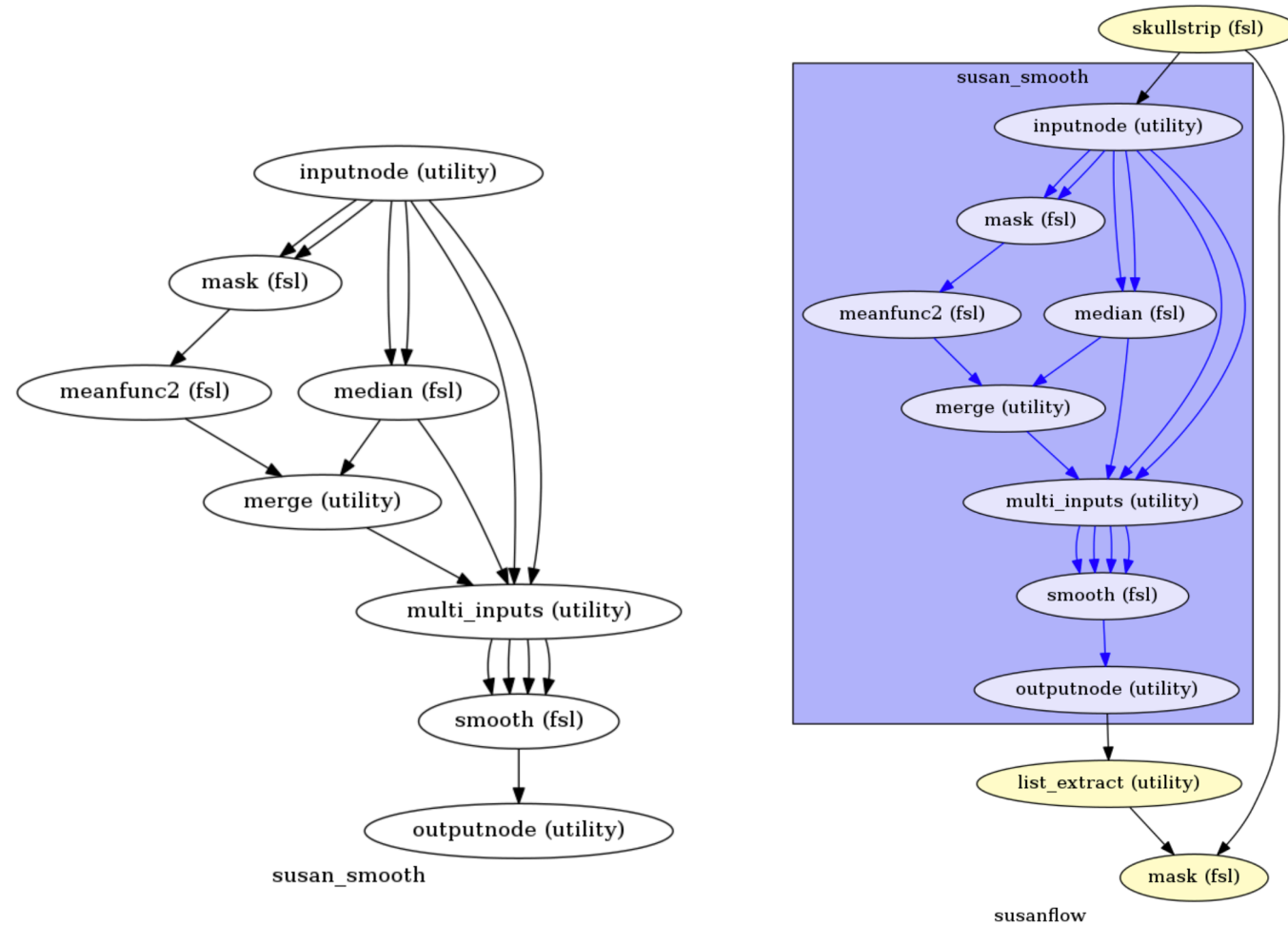
# Execute the workflow
wf.run()
```

```
180514-09:28:44,992 workflow INFO:
    Workflow smoothflow settings: ['check', 'execution', 'logging', 'monitoring']
180514-09:28:44,997 workflow INFO:
    Running serially.
180514-09:28:44,998 workflow INFO:
    [Node] Setting-up "smoothflow.smooth" in "/output/working_dir/smoothflow/smooth".
180514-09:28:45,0 workflow INFO:
    [Node] Outdated cache found for "smoothflow.smooth".
180514-09:28:45,41 workflow INFO:
    [Node] Running "smooth" ("nipype.interfaces.fsl.maths.IsotropicSmooth"), a CommandLine Interface
    with command: fslmaths /data/ds000114/sub-01/ses-test/anat/sub-01_ses-test_T1w.nii.gz -s 1.69864
    /output/working_dir/smoothflow/smooth/sub-01_ses-test_T1w_smooth.nii.gz
180514-09:28:50,11 workflow INFO:
    [Node] Finished "smoothflow.smooth".
180514-09:28:50,12 workflow INFO:
    [Node] Setting-up "smoothflow.skullstrip" in "/output/working_dir/smoothflow/skullstrip".
180514-09:28:50,40 workflow INFO:
    [Node] Cached "smoothflow.skullstrip" - collecting precomputed outputs
180514-09:28:50,42 workflow INFO:
    [Node] "smoothflow.skullstrip" found cached.
180514-09:28:50,42 workflow INFO:
    [Node] Setting-up "smoothflow.mask" in "/output/working_dir/smoothflow/mask".
180514-09:28:50,46 workflow INFO:
    [Node] Outdated cache found for "smoothflow.mask".
180514-09:28:50,52 workflow INFO:
    [Node] Running "mask" ("nipype.interfaces.fsl.maths.ApplyMask"), a CommandLine Interface with
    command: fslmaths /output/working_dir/smoothflow/smooth/sub-01_ses-test_T1w_smooth.nii.gz -mas
    /output/working_dir/smoothflow/skullstrip/sub-01_ses-test_T1w_brain_mask.nii.gz
    /output/working_dir/smoothflow/mask/sub-01_ses-test_T1w_smooth_masked.nii.gz
180514-09:28:51,134 workflow INFO:
    [Node] Finished "smoothflow.mask".
```

https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Workflow



https://miykael.github.io/nipype_tutorial/notebooks/basic_workflow.html



Preprocessing and data QC 2

Nipype (5) : Nipype Basic - Workflow

Sungwoo Lee

M.S-Ph.D. student

