

esm-runscripts

01000101
01010011
01001101

ESM-Tools

Overview

1st: Download esm-runscripts

Three possible ways:



01

```
git clone https://gitlab.dkrz.de/esm-tools.de/esm-master.git  
cd esm-master; make  
make get-esm-environment  
make get-esm-runscripts  
make get-esm-usermanual
```



02

```
git clone https://gitlab.dkrz.de/esm-tools.de/esm-master.git  
cd esm-master; make  
make get-esm-tools
```



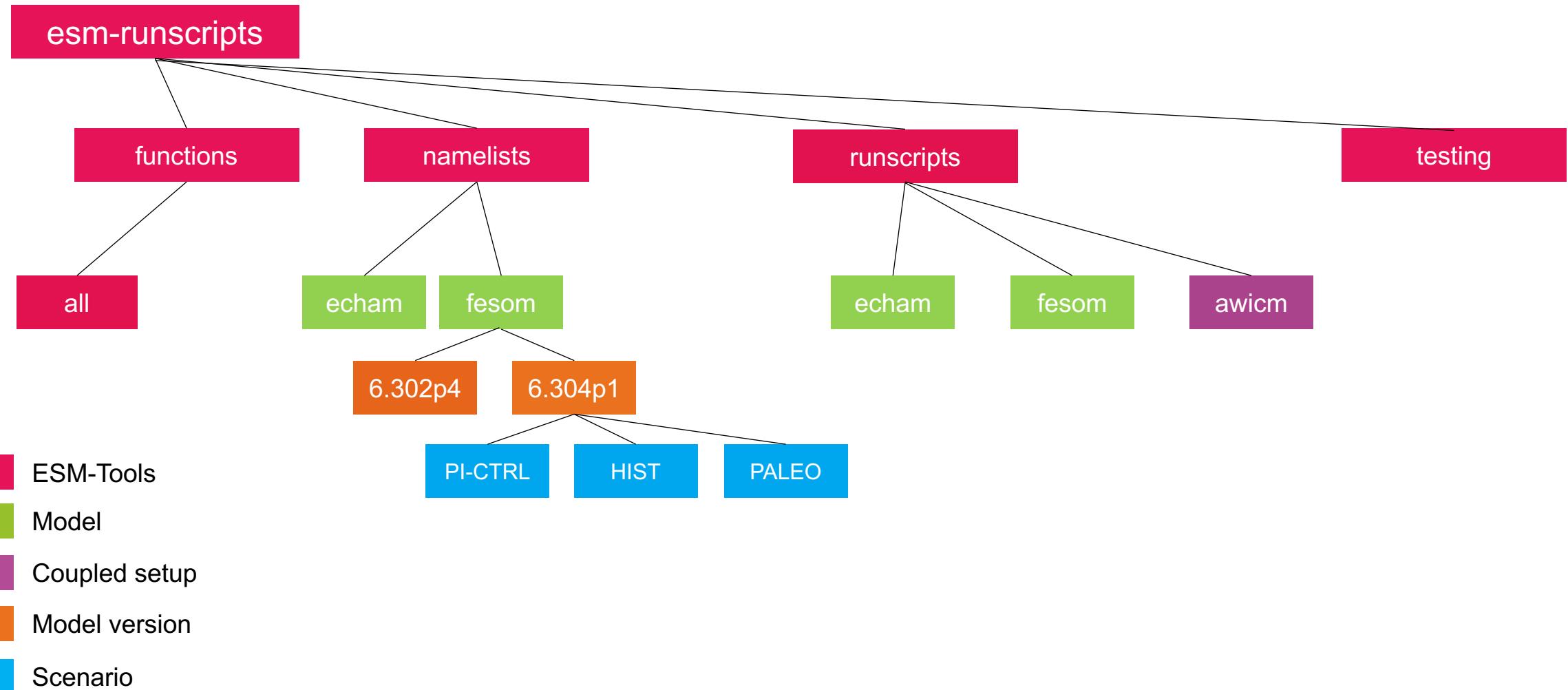
03

```
git clone https://gitlab.dkrz.de/esm-tools.de/esm-tools.git  
cd esm-tools  
../install.sh
```

01000101
01010011
01001101

ESM-Tools

esm-runscripts - folder layout



01000101
01010011
01001101

ESM-Tools

Subfolder: esm-runscripts/runscripts/awicm

The subfolder runscripts contains a collection of example runscripts, which have worked at some time and should in principle still be functional.

These need to be adapted, e.g. user defined paths, account information etc. need to be changed.

Best practice: Copy one of the runscripts somewhere else, and edit and run from there, not in this folder.

```
a270058@mlogin102% pwd
/pf/a/a270058/esm-master/esm-runscripts/runscripts
a270058@mlogin102% ls
awicm echam fesom fesom2 focioifs mpiesm nemo oifsamip pism pism_mpiesm prototype.run vilma
a270058@mlogin102% ls echam/
echam_mistral.run echam_ollie.run
a270058@mlogin102% ls awicm/
awicm-1.1_annual_initial_ollie.run      awicm-1.1_monthly_spinup_ollie.run      awicm-3.0_T511_LR_mistral.run
awicm-1.1_annual_restart_ollie.run     awicm-2.0_annual_initial_mistral.run    awicm-3.0_T511_MR_mistral.run
awicm-1.1_annual_spinup_ollie.run      awicm-2.0_annual_initial_ollie.run      awicm-3.0_ensemble.run
awicm-1.1_monthly_initial_blogin.run   awicm-3.0_T159_LR_mistral.run          awicm_OLLIE_benchmark.run
awicm-1.1_monthly_initial_juwels.run   awicm-3.0_T159_LR_ollie.run            awicm_OLLIE_initial_benchmark.run
awicm-1.1_monthly_initial_ollie.run    awicm-3.0_T255_LR_mistral.run
awicm-1.1_monthly_restart_ollie.run   awicm-3.0_T255_MR_mistral.run
a270058@mlogin102%
```

Subfolder: esm-runscripts/namelists/echam/6.304p1/PI-CTRL

```
a270058@mlogin102% pwd  
/pf/a/a270058/esm-master/esm-runscripts/namelists  
a270058@mlogin102% ls echam/  
6.3.02p4 6.3.04p1  
a270058@mlogin102% ls echam/6.3.04p1/  
1950 1percCO2 4CO2 HIST PALEO PI-CTRL  
a270058@mlogin102% ls echam/6.3.04p1/PI-CTRL/  
namelist.echam  &runcntl  
dt_start = 2285, 12, 31, 23, 52, 30  
dt_stop = 6699, 12, 31, 23, 52, 30  
putrerun = 12, 'months', 'last', 0  
lfractional_mask = .false.  
lresume = .true.  
out_datapath = './'  
out_expname = 'E280'  
rerun_filetype = 4  
delta_time = 450  
putdata = 1, 'months', 'last', 0  
npromma = 8  
lcouple = .true.  
getoceanc = 1, 'days', 'last', 0  
putoceanc = 1, 'days', 'last', 0  
lcouple_co2 = .true.  
default_output = .false.  
/
```

Default namelists are sorted in a certain way:

model / version / scenario / resolution

(those that apply)

Namelists changes can be applied in a number of ways – by changing the default file, by defining a different namelist, or by defining changes directly from the runscript.

Best practice: Small changes from the runscript, bigger changes in a new file NOT in this folder.

Subfolder: esm-runscripts/functions

esm-runscripts/functions contains dozens of SHELL and PYTHON functions, sorted into folders. A special folder is

all

that contains links to all functions. all is setup by the script set_links.sh – if something goes missing, it is save to remove

esm-runscripts/functions/all

and then call

./set_links.sh

```
a270058@mlogin102% ls
CreateTimeAxisNC.bash
amip.functions
awicm.functions
awicm3.functions
blogin.functions
bootstrap.functions
calc_date
calnoro
cmcean
cmcean-master
coupling_atmosphere2pism.functions
coupling_echam2ice.functions
coupling_fesom2ice.functions
coupling_general.functions
coupling_ice2echam.functions
coupling_ice2fesom.functions
coupling_ice2mpiom.functions
coupling_mpiom2ice.functions
coupling_ocean2pism.functions
coupling_pism2atmosphere.functions
coupling_pism2oceanc.functions
days_in_month
days_in_year
echam.functions
echam_predefined_postprocessings.functions
echam_standalone.functions
esm_calendar
fesom.functions
fesom_scalar_array_to_LonLat.py
fesom_standalone.functions
focioifs.functions
format_date
function_pperror
function_read_date
general.functions
general_clean_work.functions
general_execute_simulation.functions
general_gather_information.functions
general_helpers.functions
general_iterative_coupling.functions
general_lists.functions
general_modify_folders.functions
general_prep_folders.functions
general_prep_work.functions
general_submit.functions
general_technical_settings.functions
glogin.functions
hdmodel.functions
hlogin.functions
jsbach.functions
jsbach_init_file
jsbach_predefined_postprocessings.functions
juwels.functions
later_date
linux.functions
load_all_functions
merge_files
mistral.functions
mo_kinds.mod
mo_vegparams.mod
moab.functions
modify_restart_cover_fract.py
modify_soil_moisture.py
mpi.functions
mpiesm.functions
mpiom.functions
nemo.functions
nemo_standalone.functions
oasis3ct.functions
ocean_3eqn_ismforce.py
oifs.functions
oifsamip.functions
ollie.functions
pack_jsbach.py
pism.functions
pism_helpers.functions
pism_standalone.functions
pyfesom
rnfmap.functions
slurm.functions
time_between
time_difference
transform_ice_to_ocean.py
unpack_jsbach.py
utils
vilma.functions
vilma_standalone.functions
xios.functions
```

functions you can edit

```
#!/bin/ksh -l

vilma_set_defaults()
{
    vilma_log='vega.log'
    EXE_vilma="vega_vip.x"
    RES_vilma="512x1024"
    vilma_VERSION="I189"

    # set of parameters
    # 1.jmin and jmax, spectral resolution spherical harmonics
    vilma_jmin=0
    vilma_jmax=340

    #2. time step, if 0, read from tint in s
    TIME_STEP_vilma=${TIME_STEP_vilma:-"0.025"}
    OUTPUT_INTERVAL_vilma=${OUTPUT_INTERVAL_vilma:-"0.1"}

    #3. polynomial representation of PREM
    vilma_l_prem=1

    #4. modified 3d viscosity (0/1/2) = (no, read, adjust) MUST BE 0 (vilma_1d)
    vilma_l_visc=0

    #5. toroidal loading (don't change)
    vilma_l_toro=0

    #6. variation of rotation
    vilma_l_rot=1

    #7. load grid type (3/2/1/0)
    vilma_l_grid=2

    #8. =1, will stop after reading of environment
    vilma_l_envi=0
}

"vilma.functions" 249L, 6431C
```

You can / sometimes need to modify some of the functions. It is quite safe to edit model / setup / machine functions.

These are named in a certain way, e.g.:

echam_set_defaults
echam_prepare_forcings
echam_cleanup

These are called automatically from the general functions.

This will be much nicer once we release the python version.

01000101
01010011
01001101

ESM-Tools

functions you shouldn't edit

Some functions have grown for a long while,becoming more and more abstract and difficult to change. Especially, these are the ones starting with

general_*

Be prepared for total carnage if you try to change some of these.This is also why we recode this part in python.

Good news: No real reason to touch these.

```
general_submit() {
    next_script=$1

    eval eval eval these_submit_flags=\$submit_flags\$

    case $next_script in
        compute )
            eval nextdate=\$NEXT_DATE_\$setup_name"
            ;;
        post)
            eval nextdate=\$START_DATE_\$setup_name"
            ;;
    esac

    if [[ "x${batch_system}" == "xmoab" ]]; then
        SCRIPT_FLAGS="-v ${EXP_ID} ${cmd_line_vars} -t $next_script"
        becho "cd ${SCRIPT_DIR}; ${submit} ${these_submit_flags} ${SCRIPT_FLAGS} ${SCRIPT_DIR}/${SCRIPT_NAME}"
        cd ${SCRIPT_DIR}; ${submit} ${these_submit_flags} "${SCRIPT_FLAGS}" ${SCRIPT_DIR}/${SCRIPT_NAME}
    else
        SCRIPT_FLAGS="-e ${EXP_ID} ${cmd_line_vars} -t $next_script"
        becho "cd ${SCRIPT_DIR}; ${submit} ${these_submit_flags} ${SCRIPT_DIR}/${SCRIPT_NAME} ${SCRIPT_FLAGS}"
        cd ${SCRIPT_DIR}; ${submit} ${these_submit_flags} ${SCRIPT_DIR}/${SCRIPT_NAME} ${SCRIPT_FLAGS}
    fi
    print "starting $next_script job for $nextdate"
}
```

01000101
01010011
01001101

ESM-Tools

Adapt esm-runscripts

Adaptations needed often

1. Runscripts: Copy one to the default scripts somewhere, and adapt paths and user information. Add details for your simulation
2. Namelists: Make small namelist changes from the runscript, bigger ones by copying one of the default namelists somewhere, editing them, and then referring to them from the runscript.

Changes that can potentially be helpful for other users should be committed back...
It's an OpenSource project after all!

Adaptations needed rarely

1. Model / setup functions: E.g. defining a new scenario needs changes to the model functions
2. esm-environment: If the machine / your needs change, that should be done here.

No changes

If you need changes to the general functions – maybe ask us, info@esm-tools.net.

01000101
01010011
01001101

ESM-Tools

Basic runscript functionality

01

Prepare Input Data

Initial data, forcing data, config files (namelists) and executables are copied into distinct folders, you get a report about what isn't found.

02

Apply changes

Namelist changes, restart file manipulations etc. are performed in the experiment folder, and documented.

03

Prepare work folder

Link the previously prepared files into the work folder. When launched as a 'check' the run will stop here, and report.

04

Execute and monitor

During the actual execution of the simulation, error and graphical monitoring can provide online information about your run.

05

Cleanup output and restart files

After successful ending of the simulation, restart files and output files are sorted and copied to the folder structure.

06

Asynchronous postprocessing

In separate jobs, postprocess your data. Default postprocessing methods are pre-defined. Tar and archive.

06

Data base registering

WIP: Register your work in a personal / project data base, let others know where to find model results.

01000101
01010011
01001101

ESM-Tools

First standard run

01000101
01010011
01001101

ESM-Tools

Example:
awicm-test

1st: Download esm-tools

Three possible ways:

-  **01**
git clone <https://gitlab.dkrz.de/esm-tools.de/esm-master.git>
cd esm-master; make
make get-esm-environment
make get-esm-runscripts
make get-esm-usermanual
-  **02**
git clone <https://gitlab.dkrz.de/esm-tools.de/esm-master.git>
cd esm-master; make
make get-esm-tools
-  **03**
git clone <https://gitlab.dkrz.de/esm-tools.de/esm-tools.git>
cd esm-tools
../install.sh

01000101
01010011
01001101

ESM-Tools

2nd: Download / configure / compile the model

From the esm-master folder, type:

```
make get-awicm-test  
make comp-awicm-test
```

```
a270058@mlogin102% make get-awicm-test  
mkdir -p awicm-test; cd awicm-test; cp ..../Makefile .; cp ..../esm-master.conf .; cp -r ..../esm .;\n    make get-fesom-1.4; \  
    make get-oasis3-mct; \  
    make get-echam-6.3.04p1; \  
    rm Makefile esm-master.conf
```

```
a270058@mlogin102% make comp-awicm-test  
cd awicm-test; cp ..../Makefile .; cp ..../esm-master.conf .; cp -r ..../esm-environment .; cp -r ..../esm .;\n    make comp-oasis3-mct; \  
    sed -i '/FESOM_COUPLED/s/OFF/ON/g' fesom-1.4/CMakeLists.txt; \  
    make comp-fesom-1.4; \  
    sed -i '/ECHAM6_COUPLED/s/OFF/ON/g' echam-6.3.04p1/CMakeLists.txt; \  
    make comp-echam-6.3.04p1; \  
    mkdir -p bin; cp fesom-1.4/bin/fesom bin; cp echam-6.3.04p1/src/echam/bin/echam6 bin; \  
    rm -rf Makefile esm-master.conf esm-environment.esm; cd ..; \  
    cd .esm; source ./esm-master.functions; add_model /pf/a/a270058/to_merge/esm-master awicm-test
```

Pick a sample runscript

From the `esm-runscripts/runscripts/` folder, pick an example script, and copy it to a save place to edit it later:

```
mkdir scripts; cd scripts  
cp esm-runscripts/runscripts/awicm/awicm-1.1_monthly_initial_ollie.run awicm.run
```

There are plenty of runscripts available, depending on the machine you are about to use, whether you want monthly or yearly restarts, initial or restart from a previous run...

```
export FUNCTION_PATH=${HOME}/for_workshop/esm-master/esm-runscripts/functions/all  
export FPATH=${FUNCTION_PATH}:$FPATH  
  
machine_name="mistral"  
setup_name="awicm"  
#check=1  
  
ACCOUNT=ab0995  
compute_time="00:25:00"  
#####  
#Xsrun I know what I am doing  
  
INITIAL_DATE_awicm=2000-01-01      # Initial exp. date  
FINAL_DATE_awicm=2000-02-01        # Final date of the experiment  
  
awicm_VERSION="1.1"  
POST_PROCESSING_awicm=0  
SCENARIO_awicm="PI-CTRL"  
  
RES_fesom=CORE2  
  
MODEL_DIR_awicm=${HOME}/for_workshop/esm-master/awicm-test/  
  
echam_VERSION=echam-6.3.04p1  
LCLIBDEF_jsbach=${MODEL_DIR_awicm}/${echam_VERSION}/lctlib_nlct21.def  
CF_NAME_TABLE_oasis3mct=${MODEL_DIR_awicm}/oasis/cf_name_table.txt
```

01000101
01010011
01001101

ESM-Tools

Adapt the runscript

There are a couple of entries in the runscript you need to adapt:

FUNCTION_PATH:

ACCOUNT:

machine:

INITIAL_DATE_awicm:

FINAL_DATE_awicm:

NYEAR_awicm:

NMONTH_awicm:

MODEL_DIR_awicm:

BIN_DIR_awicm:

POOL_DIR_echam:

POOL_DIR_fesom:

MESH_DIR_fesom:

full path to esm-runscripts/functions/all

on some machines, an account (like ab0995 on mistral) needs to be given

name of the computer you want to use (ollie, mistral,...)

Start date of your simulation

End date of your simulation

Number of years per restart (default: 1)

Number of months per restart (default: 0)

full path to esm-master/awicm-CMIP6

full path to esm-master/awicm-CMIP6/bin

full path to input/forcing data pool ECHAM

full path to input/forcing data pool FESOM

full path to FESOM meshes

run a first check

Launch the runscript with:

```
./awicm.run -e mytest -c
```

This starts the run on the login node, doing everything but does not start the actual execution.

```
a270058@mlogin102% ./awicm.run -e ws2 -c
COPYING /pf/a/a270058/for_workshop/esm-master/esm-runsheets/functions/all/.../.../esm-master.conf TO EXPERIMENT FOLDER!
From now on, using: /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools/esm-master
Reading from /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools/esm-master/esm-master.conf...
COPYING /pf/a/a270058/for_workshop/esm-master/esm-environment TO EXPERIMENT FOLDER!
From now on, using: /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools/esm-environment
./set_links.sh will be run!
Conditions were:
- /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools/esm-runsheets
- reinstall_projects=0
COPYING /pf/a/a270058/for_workshop/esm-master/esm-runsheets/functions/all/.../ TO EXPERIMENT FOLDER!
>>>>>> WARNING: /pf/a/a270058/for_workshop/esm-master/esm-runsheets/functions/all/.../ not a checked-in revision (local changes)!
From now on, using: /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools/esm-runsheets
Running ./install_calendar.sh

>>>>>> WARNING: parts of the installation in /work/ab0995/a270058/esm-experiments//ws2/scripts/esm-tools are not checked-in.
Check /work/ab0995/a270058/esm-experiments//ws2/scripts/gitstatus

SCRIPT STARTING AT: Sun May 19 17:19:26 CEST 2019...
=====
GATHERING INFORMATION...
=====
All command line options: -e ws2 -c
EXP_ID=ws2
MODEL=awicm
Type of the script: compute
Tasks to be done: prep_folders modify_folders prep_work move_to_beeond execute_simulation move_from_beeond clean_work submit
t_post resubmit
Running in 'check' mode (not submitting to queue, not executing)
=====
```

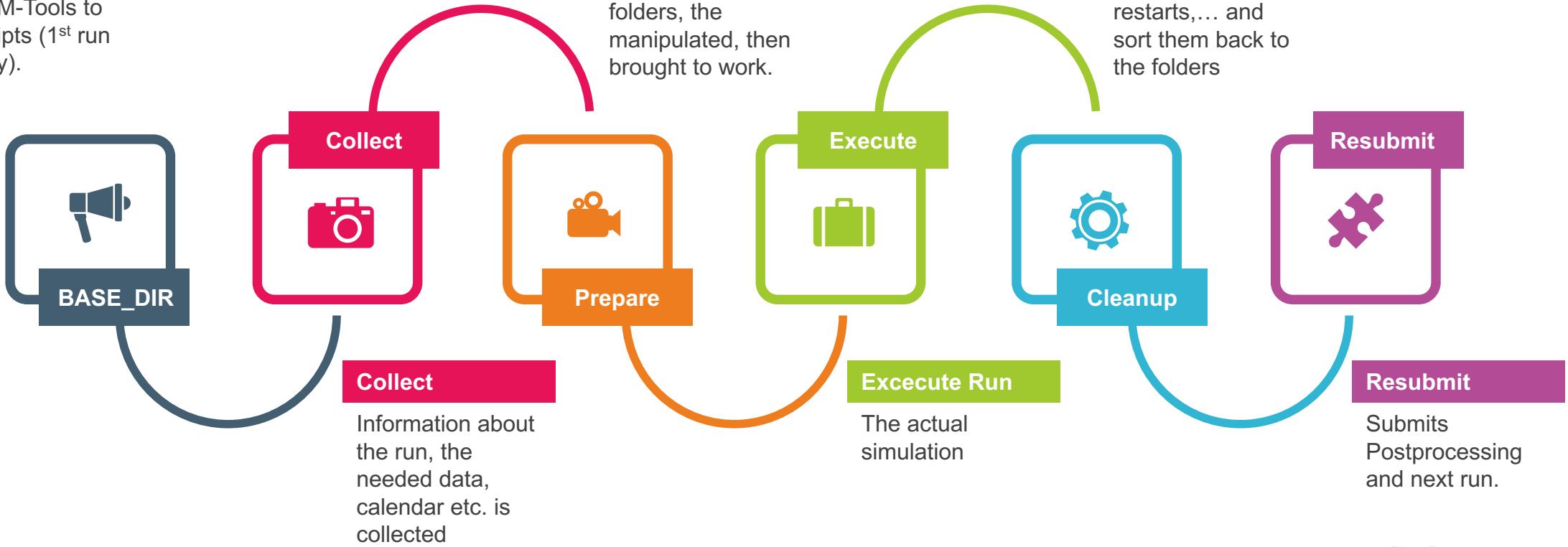
01000101
01010011
01001101

ESM-Tools

Phases of the run

BASE_DIR

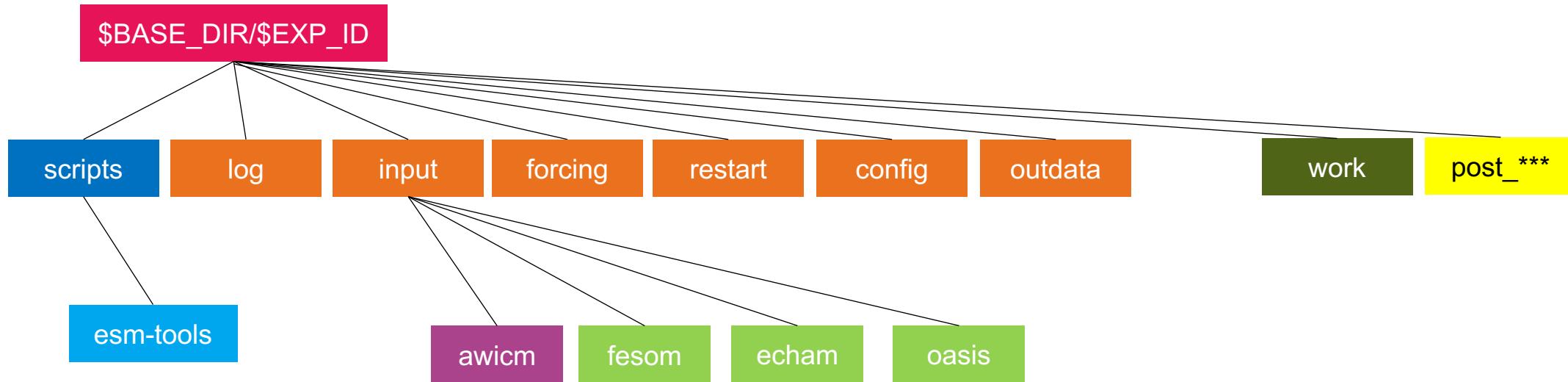
Creates main exp folder and copies ESM-Tools to scripts (1st run only).



01000101
01010011
01001101

ESM-Tools

Experiment - folder layout



[pink] Top Level for this experiment

[light green] Model

[purple] Coupled setup

[orange] Data types (sorted)

[blue] Copy of the ESM-Tools

[blue] Runscripts, main log files, date files

[dark green] Main work folder

[yellow] Maybe: One / more post folders

01000101
01010011
01001101

ESM-Tools

Main log files

- 01** **scripts/\$EXPID_compute_\$JOBID.log**
Standard output of the runscript / your job
- 02** **scripts/\$EXPID_\$setup_name_compute.log**
Main experiment tracking file, just the info
- 03** **log/awicm/*.dump**
“Dump” file containing a lot more output from the runscript, not easily readable
- 04** **scripts/gitstatus (if exists) and scripts/esm-tools/*/gitstatus**
Contains the information about the version of esm-tools you are using, also local changes
- 05** **scripts/setup*.txt and forcing/awicm/setup*.txt (and other folders)**
Keeps track about the files that were copied for use in your experiment
- 06** **scripts/missing*.txt and forcing/awicm/missing*.txt (if exists)**
Keeps track about files that should have been copied, but were not found.

Start the actual simulation

Start your simulation with:

```
./awicm.run -e myrun
```

You will see less output than from the check.

In the end, the job submits itself to the batch system.

```
STARTING COMPUTE PHASE...
=====
PREPARING NEXT compute RUN.....  

=====
Processing model echam  

Processing model fesom  

cd /work/ab0995/a270058/esm-experiments//ws1/scripts; sbatch --exclusive --account=ab0995 --partition=compute,compute2 --time=00:25:00 --mem=0 --output=ws1_compute_%j.log --error=ws1_compute_%j.log --mail-type=None --ntasks=720 --cpus-per-task=1 --job-name=ws1 /work/ab0995/a270058/esm-experiments//ws1/scripts/awicm.run -e ws1 -t compute  

Submitted batch job 16430966  

2019-05-19 16:39:25: starting compute job for 20000201  

a270058@mlogin102%  

a270058@mlogin102% !sq  

squeue -u a270058 -i 30  

Sun May 19 16:39:30 2019  

      JOBID PARTITION      NAME      USER ST          TIME  NODES NODELIST(REASON)
      16430966 compute,c    ws1    a270058 PD      0:00     15 (Resources)  

Sun May 19 16:40:00 2019
```

01000101
01010011
01001101

ESM-Tools

Congratulations!

01000101
01010011
01001101

ESM-Tools