ESM-Tools

unified and modular workflows for reproducible Earth system simulations

Deniz Ural
AWI, Climate Dynamics
Potsdam

Miguel Andrés-Martínez
AWI, Climate Dynamics
Bremerhaven

PalMod - ESM-Tools Workshop

DKRZ, 20-21 April, 2022

ALFRED-WEGENER-INSTITUT
HELMHOLTZ-ZENTRUM FÜR POLAR- UND MEERESFORSCHUNG

PAL MOD | GERMAN CLIMATE MODELING INITIATIVE

DKRZ
DEUTSCHES KLIMARECHENZENTRUM

HELMHOLTZ
RESEARCH FOR GRAND CHALLENGES

AGENDA

Day 1

12:00-12:30 - Lunch

12:30-13:00 - Introduction to the workshop and participants
ESM-Tool introduction

13:00-13:15 - Introduction to YAML

13:15-14:15 - Hands-on introduction
- Build and run FESOM

14:15-15:30 - ESM-Tools extended YAML syntax and hands-on exercises

15:30-16:00 - Coffee break

16:00-17:30 - Introduction to the workflow manager (offline coupling) and machine environments

Day 2 (Hands-on)

09:00-10:45 - Working groups
- New models into ESM-Tools
- Offline coupling
- Levante

10:45-11:15 - Coffee break

11:15-12:45 - Working groups
- New models into ESM-Tools
- Offline coupling
- Levante

12:45-13:00 - Resume

13:00-14:00 - Lunch and adjourn

OUTLINE:

1. **What are ESM-Tools?**
   - Motivation & Aim
   - Advantages
   - Supported systems
   - Repository, Documentation, Community

2. **Introduction to YAML**
   - Basic YAML Syntax
   - Brief overview of ESM-Tools Extended YAML Syntax

3. **Lets's Get Started: (Hands-on introduction)**
   - Install ESM-Tools and verify
   - Troubleshooting
   - Installed programs, command-line options
   - Brief overview of the folders and files

OUTLINE:

4 **ESM-Tools Terminology**

- Overview
- YAML Hierarchy
- Configuration files
- Runscripts
- YAML Sections
- Feature Variables
- Compilation Scripts
- .run files

5 **Hands-on Practice with FESOM 2**

- Briefing
- Install FESOM and verify
- Write our (very basic) first runscript
- Check run and verify our settings.
- Submit our simulation
- Monitor and check our simulation

OUTLINE:

6 **ESM-Tools extended YAML syntax & operations (Hands-on session)**

- Finished YAML config file
- Declaring and accessing variables
- Switches, adds, removes
- Arithmetic, Math and calendar options
- Namelist changes
- YAML hierarchy

7 **Machine files and environment_changes**

- Relevant feature variables
- environment_changes dictionaries

8 **Workflow manager and offline coupling**

- Intro to the workflow manager
- Workflow dictionary
- VILMA-PISM

Before we start:

- https://github.com/esm-tools/workshops
  - Presentations, runscripts, exercises, ...
- https://github.com/esm-tools
- https://github.com/esm-tools/esm_tools/discussions

ESM-Tools development history

Period	Developments	Team
pre-2019	• First version of the tools written in **ksh** with to support AWI couple setups **Tools development phase**	Dirk Barbi Nadine Wieters Luisa Cristini
Summer 2019-Spring 2020	• Version 3.1: Translating all the functionality to **Python** • More modular and generalized • Separation of concerns (functionality in **Python**, model-specific in **yaml**) • Support for AWI coupled systems and FOCI (GEOMAR) **Tools development phase**	Dirk Barbi Paul Gierz Nadine Wieters Luisa Cristini Sebastian Wahl
Spring 2020-Spring 2021	• Version 4.0 (April 2020): finishing off most of the Python functionality • Version 5.0 (December 2020): new models and features - AWI-ESM-2 (ECHAM6 + FESOM2) - FOCI (ECHAM6 + NEMO4) and FOCI-OpenIFS (OpenIFS43 + NEMO4) - AWI-CM-3 (OpenIFS43 + FESOM2) **Tools development phase**	Dirk Barbi Paul Gierz Miguel Andres Deniz Ural Nadine Wieters Luisa Cristini Sebastian Wahl

ESM-Tools development history

Period	Developments	Team
Spring 2021 – end 2021	• Version 6.0: - Workflow manager and **offline coupling** - **VILMA-PISM** - Stability - Focus on the end user - All packages in one single repository - CI, automatic testing - Production runs	Dirk Barbi Miguel Andres Deniz Ural Paul Gierz Sebastian Wahl Jan Streffing
2022	• Stability • Clean the Python code (refactorizations) -> adding transparency to the “blackbox” • Encourage advance users to contribute • Community building	Miguel Andres Paul Gierz Deniz Ural Sebastian Wahl Jan Streffing

Stability and user-friendliness

Stability and advance user friendly

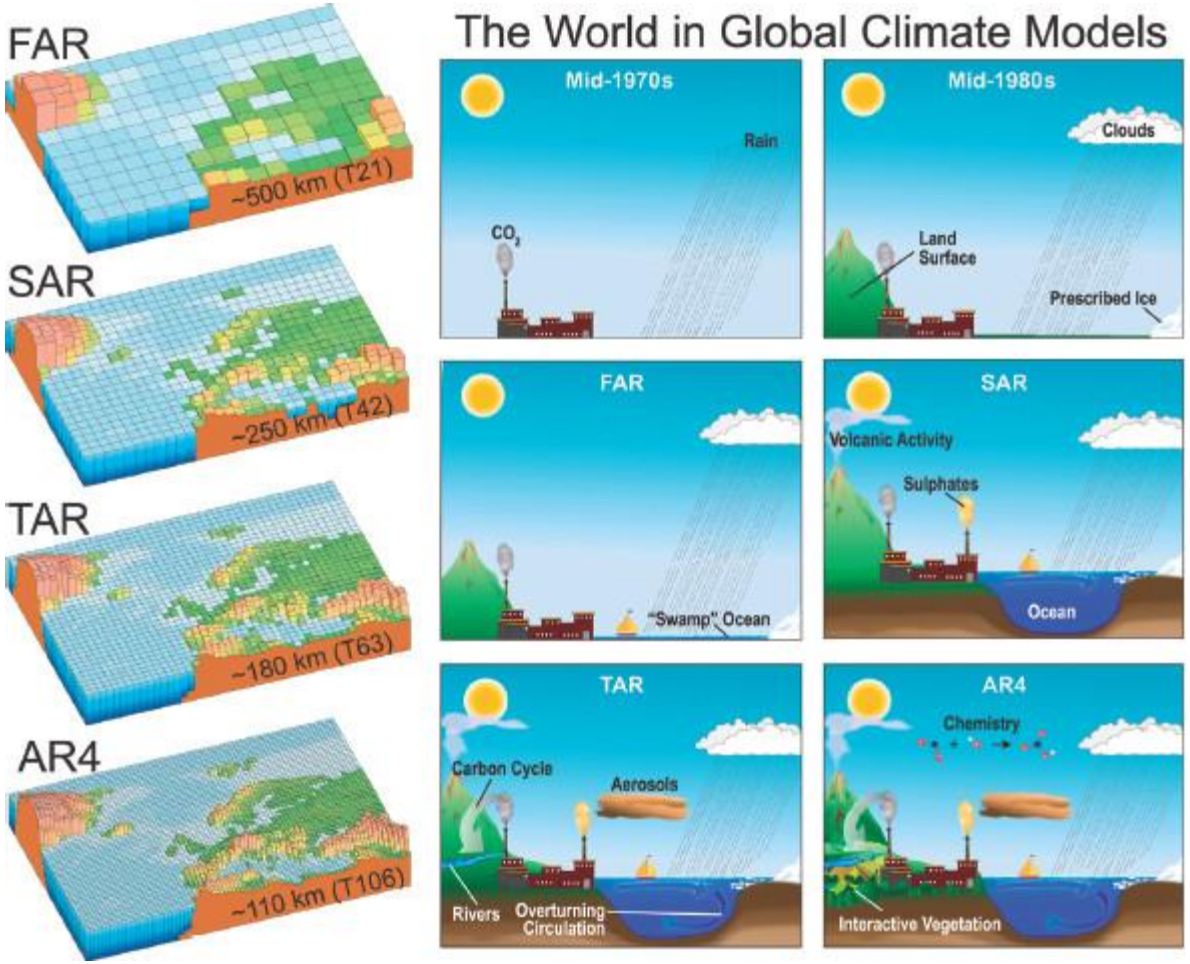
1

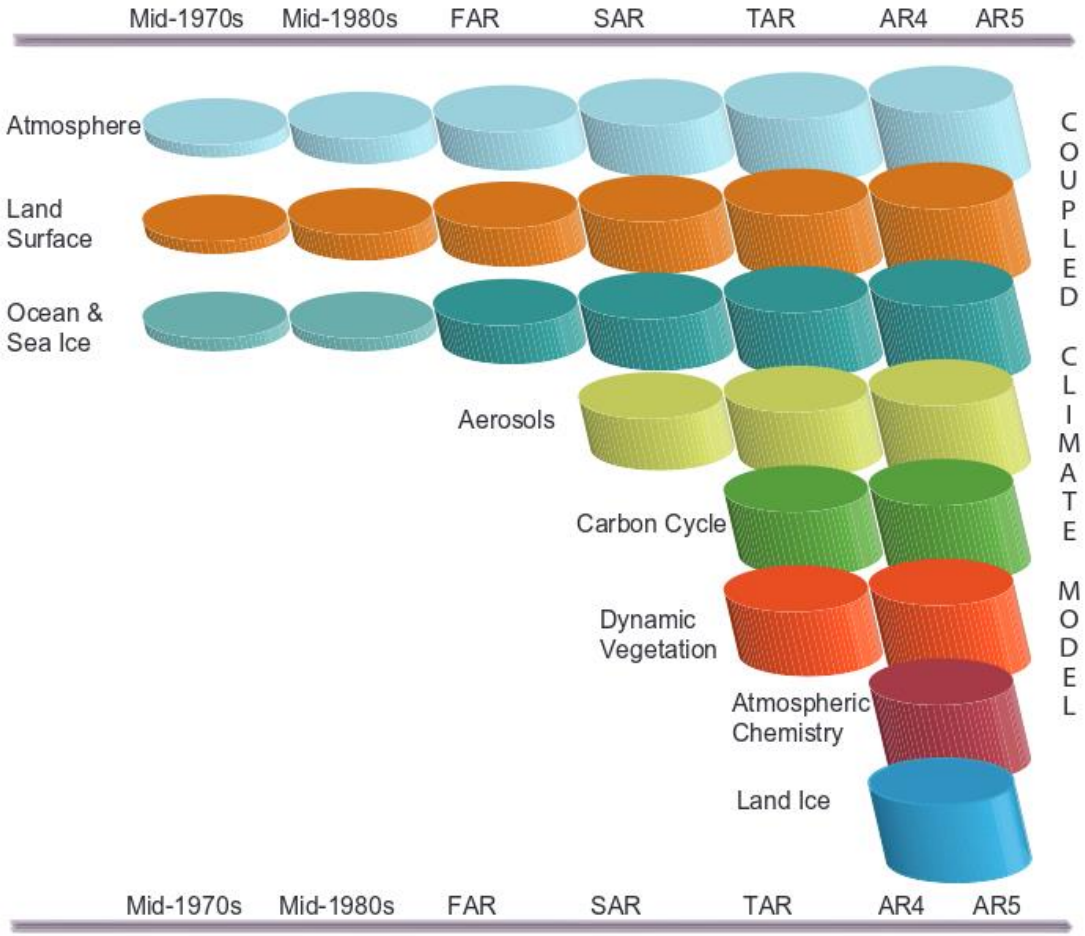
What are ESM-Tools?

Motivation & Aim
Advantages
Supported systems
Repository, Documentation, Community

Fact: Models are getting more complicated

- Great for science but hard job for the modellers





IPCC Fifth Assessment Report, 2014

1 What are ESM-Tools?

- Collection of programs to **download**, **compile**, **configure**, and **run** different Earth system models (ESM)
- **Standalone** Atmosphere, Ocean, Geo-Biochemistry, Hydrology, Sea-Ice and Ice-sheet models as well as **coupled** systems
- Researchers should focus on **science** and less on technical details
- Provide a **common infrastructure** for
  - Models and coupled systems
  - HPC environment
  - Setup and run model experiments
  - Consistent file / directory structure

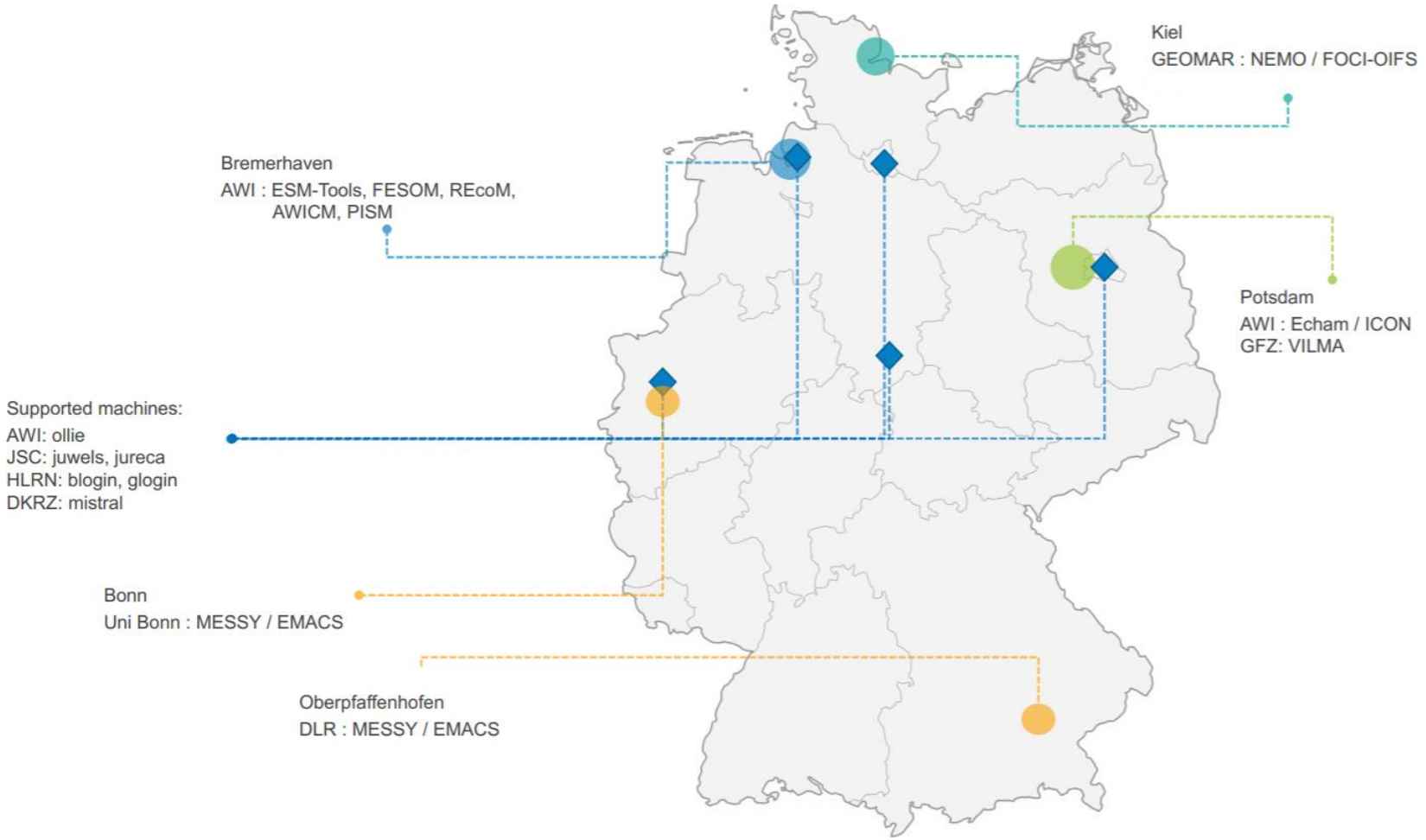
1 Why do we need ESM-Tools?

- **Target audience:** Earth System modellers working on HPC machines
- ESM are **complex** softwares that require **technical** knowledge
- **Build** is difficult:
  - Many different models & different build systems & different configurations
  - Different HPC and batch systems
- **Setup** & **Run** are difficult:
  - Complex configurations & Couplings
  - Requires and generates many files
  - Requires a consistent directory structure, CMORization
  - Automatization: Repeating the same simulation multiple times
  - Reproducibility

Less technical demand → more time for science & research

1 Supported Models & Couplings, Partners

Coupled Systems	Components
AWI-CM1	**ECHAM**
AWI-CM3	**FESOM**
AWI-ESM-2.1	ICON
FESOM-REcoM	**NEMO**
FOCI	**OASIS3MCT**
FOCI-OIFS	**OpenIFS**
	PISM
	REcoM
	RNFMAP
	VILMA
	xios
	YAC



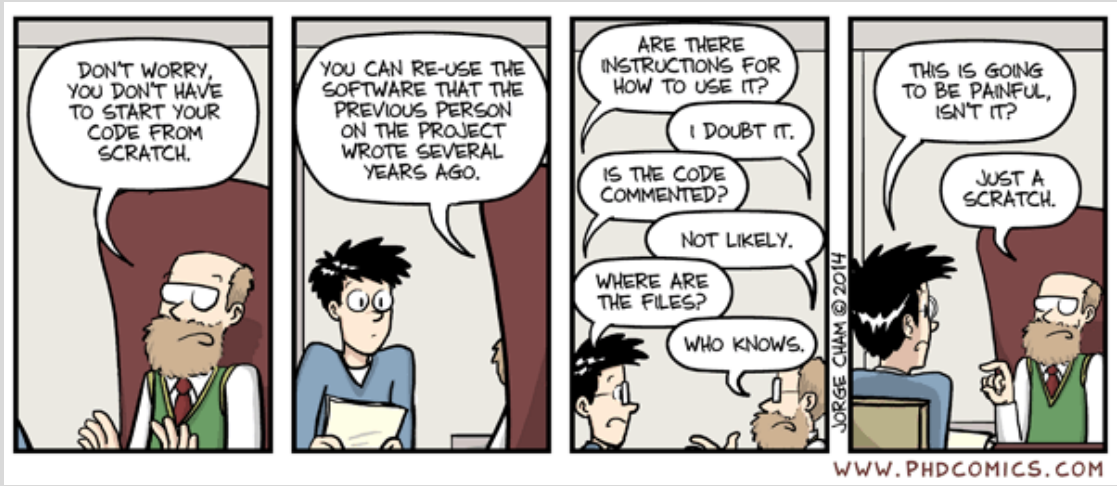
Need for the high-quality research software

BETTER SOFTWARE

VS

BETTER RESEARCH

www.software.ac.uk



Prof. Carole Goble
Software Sustainability Institute
https://ieeexplore.ieee.org/document/6886129

http://phdcomics.com/comics/archive/phd031214s.gif

Old Workflow

VS

ESM-Tools workflow

1. Obtain the model source code (usually a tar ball)
2. Build the model
    1. `configure`
    2. `make`
    3. ❌ **FAIL:** Read the HPC documentation and repeat (libraries, compilers, modules, ...)
3. Prepare the data folders (input, boundary conditions, output, ...)
4. Setup the namelist for the models
5. Submit your job to the HPC system
    - ⚠ **Warning:** environment mismatch
6. Resubmit / Iterative coupling
7. Move the data to the storage disk
8. Postprocessing of the results

Repeat the whole process for the next run or write a shell script for automatization.

- **Obtain** and **build** the model code (from a repository)
    - ✅ **Uniform** environment for both installation and running → guaranteed integrity.

```
esm_master install-awicm-2.0
```

- Prepare **YAML** based runscript
- [OPTIONAL] **Check** if your run would run successfully:

```
esm_runscripts my_awicm_runscript.yaml -e my_first_test -c
```

- **Submit** your job to the system:

```
esm_runscripts my_awicm_runscript.yaml -e my_first_test
```

- Monitor your log files
- **Postprocess** the results (esmviz, in progress)

Old Workflow

VS

ESM-Tools workflow

```
echam_prepare_forcing()
{
    # forcing

    if [[ "v$setup_name" = "vecham_standalone" ]]; then

        case $SCENARIO_echam in
            1850 | PI-CTRL*)
                add_to ${echam_INPUT_DIR}/${RES_echam}/${RES_echam}${OCERES_echam}_piControl-LR_sst_1880-2379.nc unit.20
                add_to ${echam_INPUT_DIR}/${RES_echam}/${RES_echam}${OCERES_echam}_piControl-LR_sic_1880-2379.nc unit.96
                ;;
            HIST )
...
...

for ((yr = YR0_echam + -2; yr <= YRN_echam + 2; ++yr)); do
    if [ $yr -le 1849 ] ; then
        eval add_to ${echam_INPUT_DIR}/${RES_echam}/ozone/$ozonefile_1850 ozon$yr

    elif [ $yr -le 2014 ] ; then
        eval add_to ${echam_INPUT_DIR}/${RES_echam}/ozone/$ozonefile_hist ozon$yr

    else
        eval add_to ${echam_INPUT_DIR}/${RES_echam}/ozone/$ozonefile_scen ozon$yr
    fi

    if [ $yr -le 1849 ] ; then
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_ir_${RES_echam}_1850.nc
strat_aerosol_ir_${yr}.nc
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_sw_${RES_echam}_1850.nc
strat_aerosol_sw_${yr}.nc

    elif [ $yr -le 2024 ] ; then
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_ir_${RES_echam}_${yr}.nc
strat_aerosol_ir_${yr}.nc
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_sw_${RES_echam}_${yr}.nc
strat_aerosol_sw_${yr}.nc

    elif [ $yr -gt 2024 ] ; then
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_ir_${RES_echam}_2024.nc
strat_aerosol_ir_${yr}.nc
        add_to ${echam_INPUT_DIR}/${RES_echam}/volcano_aerosols/strat_aerosol_sw_${RES_echam}_2024.nc
```

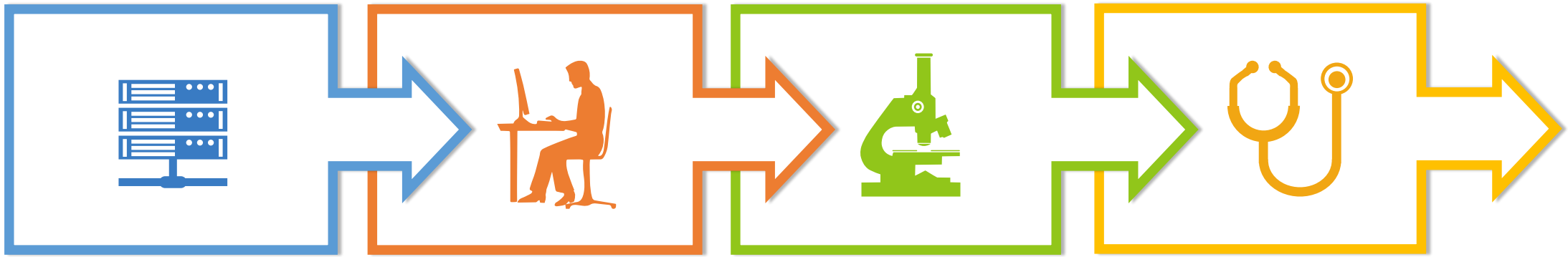
... and many more

```yaml
general:
    setup_name: "awicm"
    compute_time: "00:15:00"
    initial_date: "2000-01-01"
    final_date:   "2000-02-29"
    base_dir: "/work/ollie/dural/sample_work_dir/"
    nmonth: 1
    nyear: 0

awicm:
    version: "CMIP6"
    postprocessing: false
    scenario: "PI-CTRL"
    model_dir: "/work/ollie/dbarbi/modelcodes/awicm-CMIP6/"

fesom:
    pool_dir: "/work/ollie/pool/FESOM/"
    mesh_dir: "/work/ollie/pool/FESOM/meshes_default/core/"
    restart_rate: 1
    restart_unit: "m"
    restart_first: 1
    further_reading:
        - "fesom_output_control.yaml"
```

1 Advantages of ESM-Tools

- **Automation:** minimal manual interaction
- **Uniform** (same structure)**, Data Integrity → Reproducible**
- **Portability:** across different supported HPCs
- **Abstraction:** configuration (YAML) and operations (Python) are separated
- **Stateful:** simulation configuration is stored
- **Modular & Extendable:** easy to implement a new model and coupled setup or user plugins

1 Who benefits from ESM-Tools



System Admins

Standardized Compile and Runtime Environments means fewer needed software packages. Deploy optimal machine settings or new modules easily.

Model developers

Organize your developments, **deploy** them on different machines. Co-work with other institutes on the same code.

Modellers

Run your simulations in an **easy** and **unified** way, independent of the model and hardware. Have lots of functions with a few lines of runscript.

Model supporters

Solve problems **once**, not over and over again. Deploy **bugfixes** / new hardware configurations **quickly** to all users. Same experiment layout also means less context switching.

1 Technical reasons for using ESM-Tools

For Users

- (extended) YAML syntax is **easy** to read
- Sample runscripts are already available
- Well **maintained**
- Issues on GitHub (and we will take care of them)
- **Updated** regularly
- Portable & **Tested**
- **Documentation** (sphinx, readthedocs)
- **Workshops**

For Developers

- python powered
- Easier to read / write (compared to 

)
- OOP, High level data structures
- Easier to **debug** (via pdb, ipdb)
- Configuration files are **inherited**
- Hosted on GitHub & robust branching model
- **Open-source** development is encouraged
- [In progress] CI/CD, DevOps, Automated **tests**

1 Recap: Aim & Motivation

What ESM-Tools are / do

- A **unified** infrastructure for ESM modelling
- Build the models **without** knowing the details of the HPC system
  - great for education, new colleagues
- **Run** your simulation as easy as possible
- **One interface:** standardize the modelling process for all of your models
  - One software to rule them all
- Provide **easy** to read/write **YAML** based configuration
- Generate a **log** documentation, easy **monitoring**
- **Organize** files & directories (eg. input, forcing, output, log, executables, ...), **archieve** or **clean** your simulations.

What ESM-Tools are / do not

- A new model
- A new coupler
- A new imperative programming language
- Change the model code / build process

1 Contact & Community

https://github.com/esm-tools

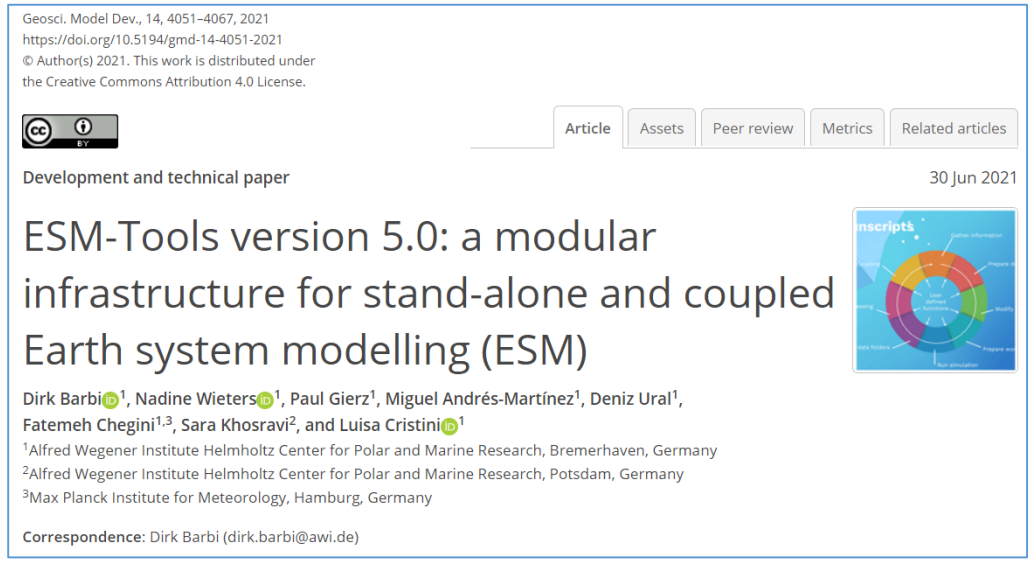
- ✓ https://github.com/esm-tools/esm_tools/discussions
- ✓ https://github.com/esm-tools/esm_tools/issues

https://esm-tools.readthedocs.io

https://www.esm-tools.net

@ToolsEsm

YouTube **ESMTools**



https://gmd.copernicus.org/articles/14/4051/2021/

Barbi *et. al.*, GMD (2021): https://doi.org/10.5194/gmd-14-4051-2021

Zenodo: https://doi.org/10.5281/zenodo.5787476

1 Community: Issues and Discussions

- **Open** to everyone
- **Single** platform
- **Searchable**
- **Agile**