



CRC1502
DETECT

<https://sfb1502.de/>



Status of TSMP2 by DETECT/Z04

2025-05-14 | Stefan Poll^{1,2,3}, Marco van Hulsen⁴, Johannes Keller^{1,2,°}, Paul Rigor^{1,2,°},
Daniel Caviedes-Voullième^{2,3,°}, Klaus Goergen^{1,2} (Z04 and °affiliated)
Under the auspices and with support of groups from Stefan Kollet^{1,2} and Harrie-Jan Hendricks-Franssen^{1,2}
Supported by Kaveh Patakchi Yousefi^{1,2}, Niklas Wager^{1,2,*}, Carl Hartick^{1,2,*}, Muhammad Fahad^{1,2},
Ana Gonzalez-Nicolas^{2,3}

¹Institute of Bio- and Geosciences (IBG-3, Agrosphere), Research Centre Jülich (FZJ), Jülich, Germany

²Centre for High-Performance Scientific Computing in Terrestrial Systems (HPSC TerrSys), Geoverbund ABC/J, Jülich, Germany

³Simulation and Data Lab Terrestrial Systems (SDLTS), Jülich Supercomputing Centre (JSC), Research Centre Jülich (FZJ), Jülich, Germany

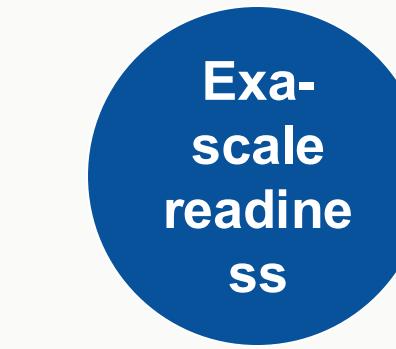
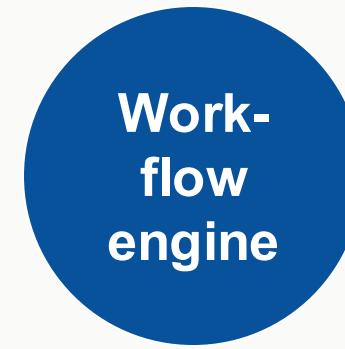
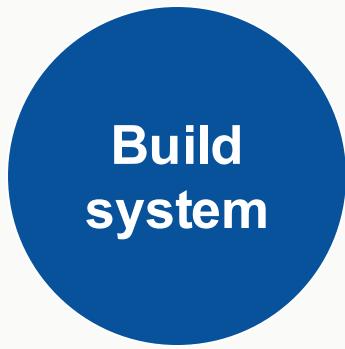
⁴High-Performance Computing and Analytics Lab (HPC/A-Lab), University of Bonn, Bonn, Germany

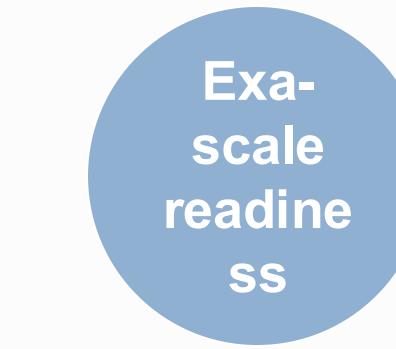
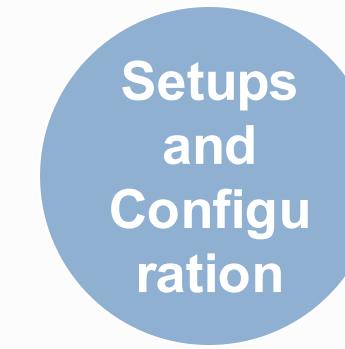
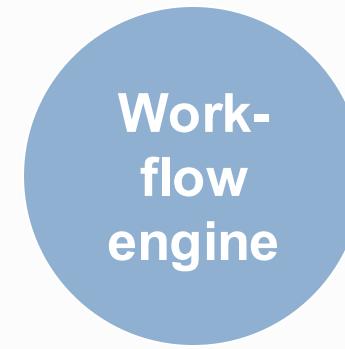
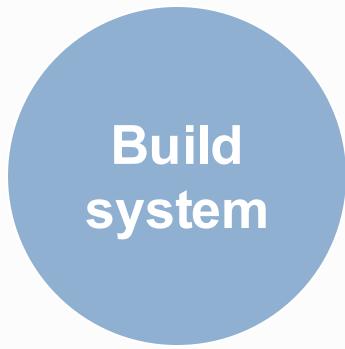
*New affiliation



CRC1502
DETECT

<https://sfb1502.de/>





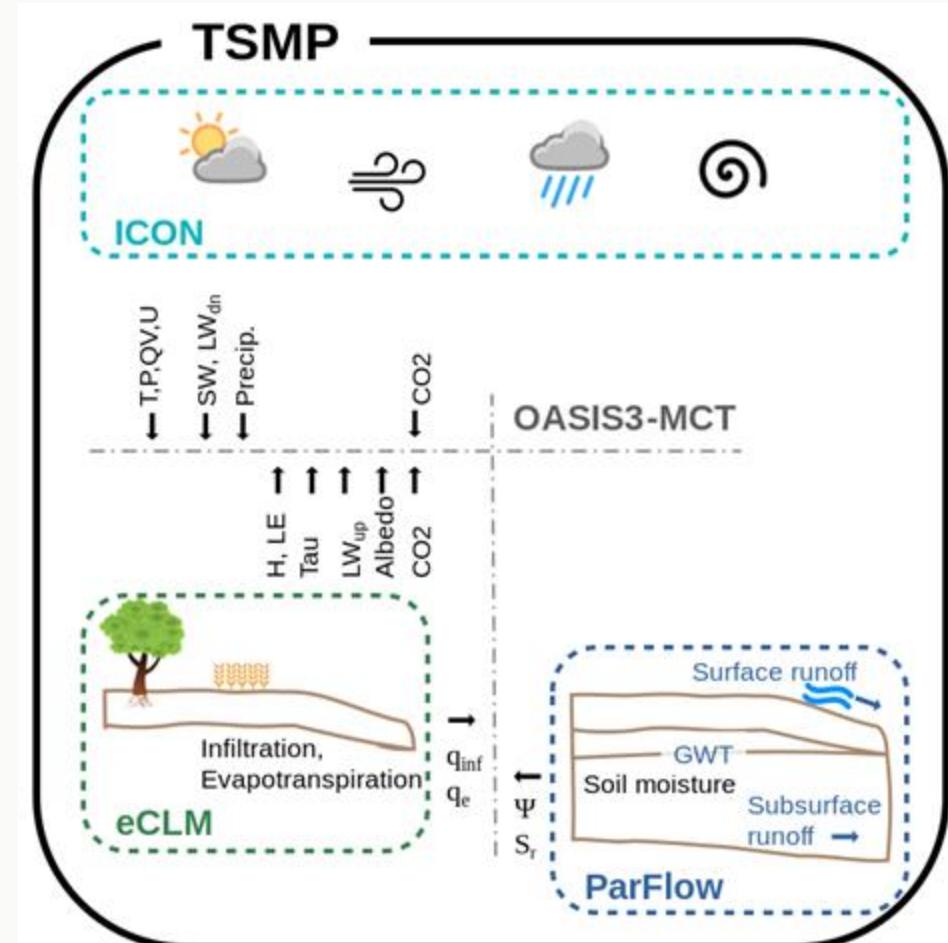
Goals and Overview

TSMP overview, coupling, goals in DETECT I



TSMP2 framework – Overview

Terrestrial Systems Modelling Platform v2 (TSMP2)



- Modular coupling design (MPMD + HC)
- Open source (framework + components)

ICON – v2024.07

Zängl et al.,
2015
Atmospheric
model

**eCLM – Fork of
NCAR's CLM5.0**

based on
Lawrence et al.,
2019
Land-surface
model

ParFlow – v3.13

Kollet and Maxwell,
2006; Kuffour et al.
2020

Integrated
hydrological
model

OASIS3-MCT – v5.0

Craig et al., 2017
External coupler



TSMP framework – Atmosphere-Land coupling

Explicit coupling approaches after Polcher et al. (1998)

TSMP1

Exchange coefficients approach

- Coupling via exchange coefficients ($C_{\theta,q,M}$) and surface variables
- Calculation of surface fluxes

Surface flux calculation in ICON

$$H_0 = -\rho c_p C_\theta |v_h| \Delta T$$

$$(L_v E)_0 = -\rho L_v C_q |v_h| \Delta q_v$$

$$M_{i,0} = -\rho C_M |v_h| \Delta u_i$$

TSMP2

Fixed fluxes approach

- Coupling via turbulent fluxes
- Direct usage of surface fluxes



TSMP2 framework – Land-Subsurface coupling

Explicit coupling approach; coupling via Richards equation

Surface and subsurface component models share the same vertical discretization

- Infiltration and root zone transpiration (eCLM) show up as Q
- Pressure head (ParFlow) replaces soil matric potential (eCLM)
- ParFlow overrides soil hydraulic properties specified in eCLM
- Soil water is computed to ParFlow and then passed to eCLM

Richards equation:

$$S(p)S_s \frac{\partial p}{\partial t} - \frac{\partial(S(p)\rho(p)\phi)}{\partial t} - \nabla \cdot (\mathbf{K}(p)\rho(p)(\nabla p - \rho(p)\vec{g})) = Q$$

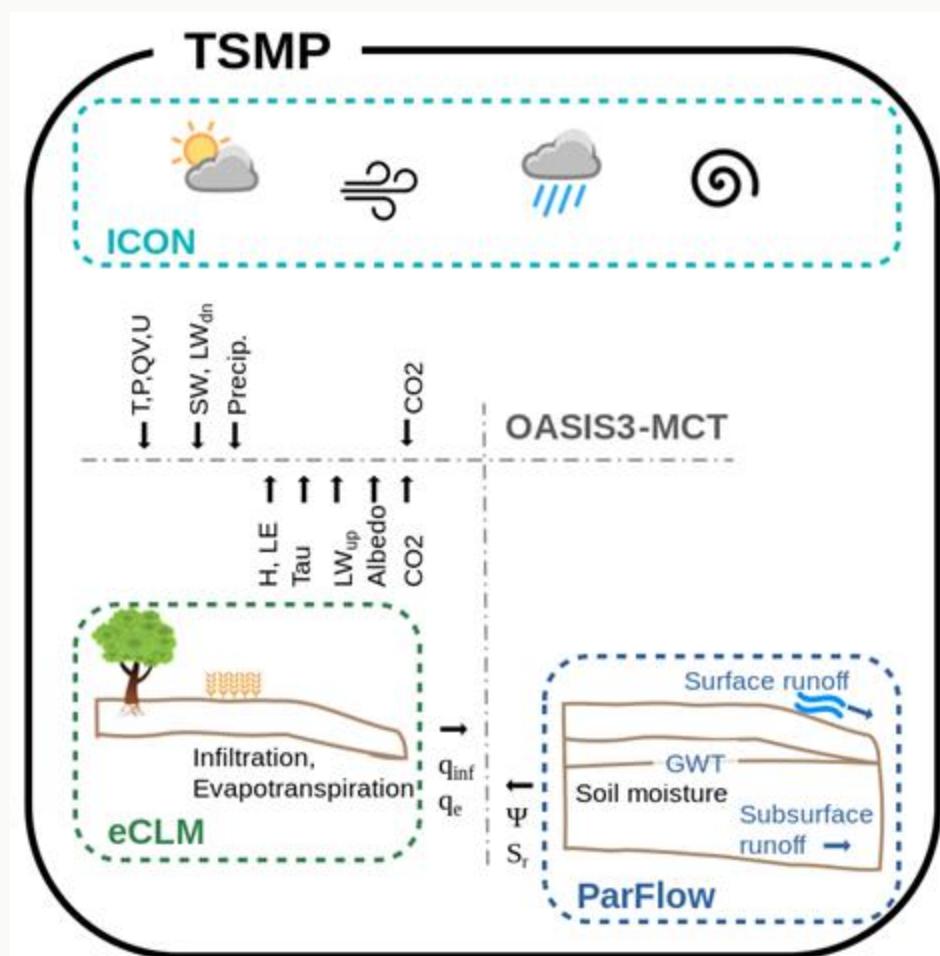
where:

- | | |
|---|------------------------|
| p – pressure head | ρ – water density |
| S – water saturation | g – gravity |
| ϕ – porosity | |
| S_s – specific storage coefficient | |
| $\mathbf{K}(p)$ – hydraulic conductivity tensor | |



TSMP2 framework – Context and overall usage

Terrestrial Systems Modelling Platform v2 (TSMP2) in DETECT



- The Integrated Modelling System (IMS) is important for hypothesis testing in DETECT CRC
- IMS can simulate feedbacks and interactions in groundwater-land-atmosphere system
- Fully coupled TSMP2 is usable, all components and software tools in the workflow are ready
- Increase user friendliness
 - Front end in the building system
 - Introducing a runtime environment
- TSMP2 component models in use by DETECT WPs; the fully coupled is not used yet, though there is potential
- Fully coupled TSMP1 (COSMO-CLM3.5-ParFlow) is “workhorse” for hypothesis testing, e.g. D02



TSMP2 fully coupled simulation experiments, towards RESM

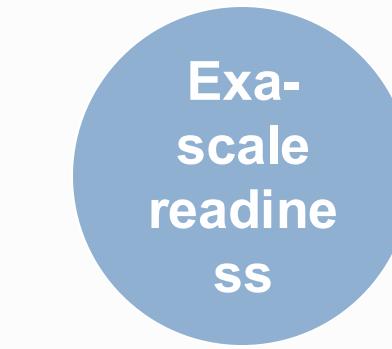
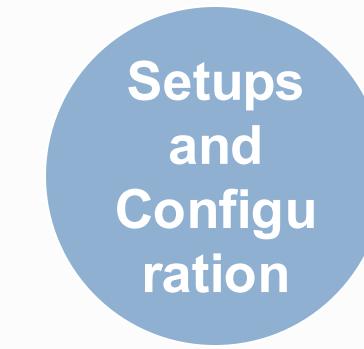
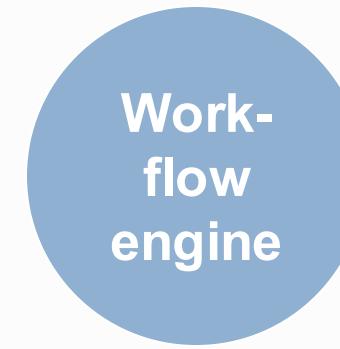
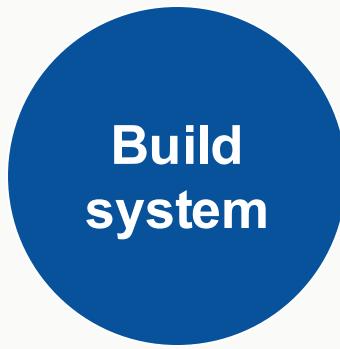
Goal: Must have TSMP2 fully usable for DETECT / Phase 2 and published beforehand

Workflow is ready: simulation setup (HPC system, software environment, built system, domain, external parameter fields), model configuration, compute configuration, initial conditions, automated workflow engine (BC pre-processing, run-control / -monitoring, (post-)processing, archiving), dissemination (full data lifecycle)

Planning – once data is produced more analyses are possible, Z04 will conduct the TSMP2 3km baseline run:

1. 2025/Q1-Q2: TSMP2 introduction paper (e.g., GMD(D)), using 12km and idealized tests; proof of concept; baseline paper for TSMP2
2. 2025/Q2-Q3: TSMP2 high resolution, coupled, pan-EU frontier experiment presentation and evaluation paper (e.g., GRL), short test / annual cycles / individual weather events / multi-year eval runs

Baseline setup: 1hr ERA5-driven, direct nest, CLMcom ICON configs, planned/needed: 2000-2020 (time-slices), transient GHGs and aerosols, static land cover, no HWU, needs IC from ParFlow standalone run EUR-0275, run “in the background”



Build System

Developments in build system, component models and beyond



Building within TSMP2 framework

New TSMP2 frontend for more convenient building of model components



Build your model in just two steps:

- 1) Clone repository: git clone <https://github.com/HPSCTerrSys/TSMP2>
- 2) Build model components: ./build_tsmp2.sh --ICON --eCLM --ParFlow

Supported machines (by default):

- JUWELS, JURECA-DC, JUSUF
- JEDI / JUPITER
- MARVIN
- Ubuntu

Multiple Environments possible

- Intel, GNU

The screenshot shows the GitHub repository page for 'HPSCTerrSys/TSMP2'. The repository has 12 stars and 2 forks. It contains several branches: master, v1.0, v1.1, v1.2, v1.3, v1.4, v1.5, v1.6, v1.7, v1.8, v1.9, and v1.10. The 'master' branch is selected. The repository has 12 commits, all pushed 19 hours ago. The commit history includes: 'Promote for TSMP2 (x64)', 'Built (AMSG-MCT by default instead of using prebuilt lib...) (x64)', 'Promote for TSMP2 (x64)', 'Add MIT License in LICENSE.md and CONTRIBUTORS.md...', 'Add MIT License in LICENSE.md and CONTRIBUTORS.md...', and 'Promote for TSMP2 (x64)'. The repository has 1 release and 0 packages. It also lists contributors and languages (CMake 100%, Shell 100%). A 'Quickstart' section provides instructions for running the build script.



Model improvements & upgrades

Optimisation of the coupling

Update of component models

- i) ICON: update to open-source release, synced with icon/icon in background
- ii) eCLM: update procedure in planning
- iii) ParFlow: follows upstream

ICON-eCLM coupling:

- i) eCLM first receive from ICON, then read streams → better flow of data
- ii) Do not couple lateral boundary points (halo-region)



TSMP2-PDAF: Data assimilation with TSMP2

Collaborative development environment

Build

- TSMP2 repository:
<https://github.com/HPSCTerrSys/TSMP2/>
- New frontend for simplified user experience:
`./build_tsmp2.sh --eCLM --PDAF`

Documentation

- https://hpscterrsystech.github.io/TSMP/content/setup_tsmp/input_enkfpf.html

Issues

- In-repo (TSMP2, eCLM, PDAF,...)
- DETECT (https://gitlab.jsc.fz-juelich.de/detect/detect_z03_z04/discussions/-/issues)

The screenshot shows two pages from the TSMP2 documentation. The left page is the 'Control file enkfpf.par' page, which includes a sidebar with sections like 'INTRODUCTION', 'GETTING STARTED', 'TECHNICAL NOTE TSMP', 'BUILDING TSMP', 'SETTING UP TSMP', and 'Running TSMP-PDAF'. The main content discusses the 'enkfpf.par' file, its purpose, and its structure. It lists four categories: [PF], [CLM], [COSMO], and [DA]. An example code snippet for 'enkfpf.par' is provided, showing parameters for ParFlow, CLM, COSMO, and DA. The right page is the 'Running TSMP-PDAF' page, which contains a table of contents for various input files and a detailed view of the 'Control file enkfpf.par' section. This section lists numerous parameters such as problemname, nprocs, starttime, dt, endtime, sendtime, updateflag, gmasking, paramupdate, paramupdate_frequency, dampingfactor_param, dampingFactor_state, damping_tswitch_sm, aniso_perm_y, aniso_perm_z, aniso_use_parflow, printensemble, and t_printensemble.

TSMP2-PDAF: Data assimilation with TSMP2

Status and Plans

Status

- Recent developments:
 - New observation types in TSMP-PDAF: Snow-Depth / SWE, LAI (point-scale), LST
 - Local filters (e.g. LESTKF) for eCLM-PDAF

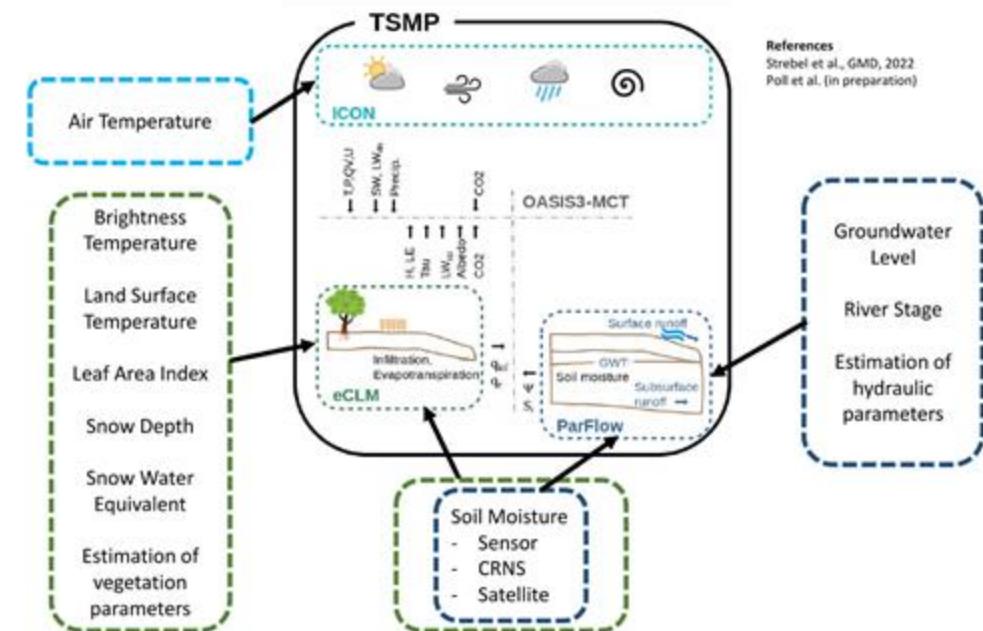
Plans for 1st phase of DETECT

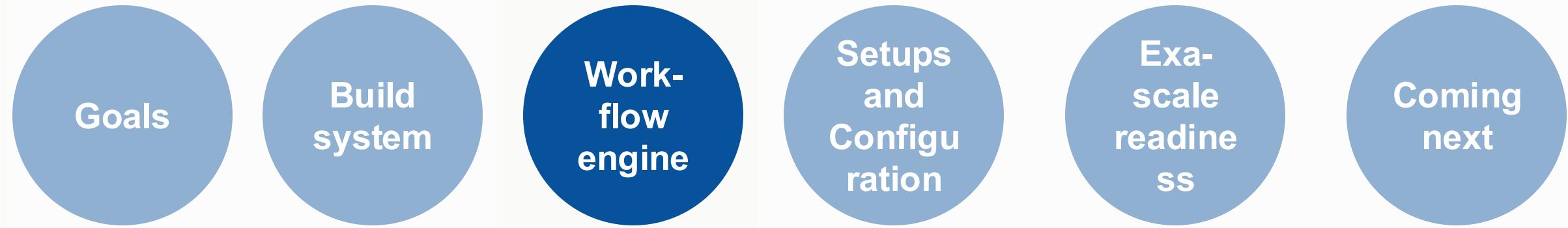
- OMI-interface, GRACE-DA currently in University Bonn branch (Yorck), include into TSMP-PDAF
- Fully functional (assimilation into point-scale and distributed simulations)
 - LST-DA
 - Snow-DA (SWE / Snow-Depth)
 - LAI-DA

Proposals for 2nd phase of DETECT (new in bold)

- **Flexible multivariate assimilations** of all observation types, i.e. SM, in-situ ET, GRACE data, LST, Snow variables, LAI
- **Hypothesis testing** with TSMP-PDAF
- **Snow-cover DA**

TSMP-PDAF Observations







Runtime environment – TSMP2 workflow engine

Convenient way to perform simulations

Why using TSMP workflow engine?

- Simulation control and reproducibility
- Pre-processing, simulation, post-processing, visualization



Features

- Flexible usage of any model component and combination in TSMP2 realm
- Usable for component models standalone and coupled systems
- For test, production and sensitivity runs
- Download and “direct” start

Workflow engine as the skeleton

- Is not functional in itself
- “Knows” where to set the parts



Technicalities

- Version controlled (git based)
- Git repository at github
- Shell-based / simple structure
- Choice of experiment based on submodules

Git submodules make it complete

- for model source code, model namelist
- for static fields
- for post/pre-processing scripts
-





TSMP2 workflow engine

Structure of TSMP2 WFE

TSMP2_wfe/

- |--- ctl -> managing the simulation
- |--- nml -> model namelists
- |--- run -> number crunching
- |--- dta -> data
- |--- src -> source codes

Two configure files to manage configuration / setup

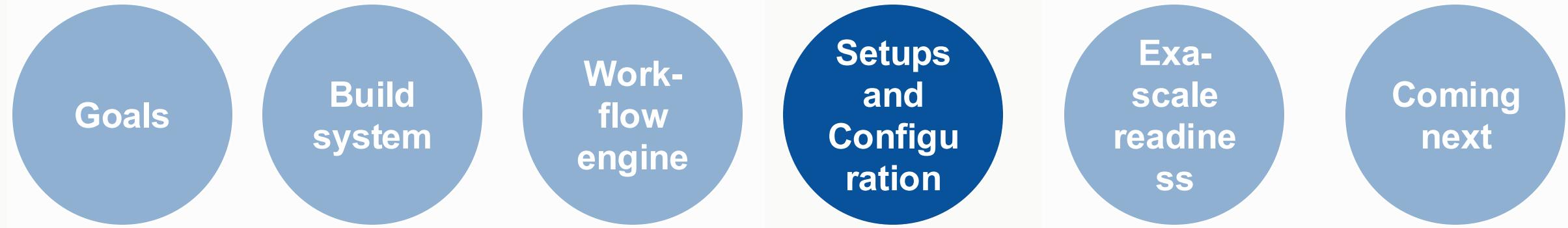
- |--- master.conf -> general settings
- |--- expid.conf -> model specifications

master.conf (excerpt)

```
# main settings
MODEL_ID=ICON-eCLM-ParFlow #ParFlow #ICON-eCLM #ICON-eCLM-ParFlow #ICON
EXP_ID="eur-11u"
CASE_ID="" # identifier for cases
conf_file="" # configure file for cases, also possible ${EXP_ID}.conf

# main switches (PREprocessing, SIMulations, POST-processing, VISualisation)
lpre=( false false false ) # config, run, cleanup
lsim=( true true true ) # config, run, cleanup
lpos=( false false false ) # config, run, cleanup
lvis=( false false false ) # config, run, cleanup

# time information
cpltsp_atmsfc=900 # coupling time step, atm-sfc, eCLM timestep [sec]
cpltsp_sfcss=900 # coupling time step, sfc-ss, ParFlow timestep [sec]
simlength="1 day" #"23 hours"
startdate="2017-07-01T00:00Z" # ISO norm 8601
inidate=${startdate} # ISO norm 8601, fix for entire simulation
numsimstep=1 # number of simulation steps, simulation period = numsimstep * simlength
```





Generate of new domains

eCLM static file generator

- Creation of new static parameters
- Make consistent static fields across TSMP2 compartments
- Creation of eCLM/CLM5 atmospheric forcing files

The screenshot shows a GitHub repository page for 'eCLM_static-file-generator'. The repository has 53 commits from 'mvhulten' over the last month. The commits are listed with their descriptions and timestamps. The repository has 5 forks and 0 stars. It includes sections for 'About', 'Releases', 'Packages', and 'Contributors'.

Commits:

- main · mvhulten · 53 Commits · last month
- gen_domain_files/src · Tested and improved scripts and documentation (#11) · 2 months ago
- mkdirconfig · Tested and improved scripts and documentation (#11) · 2 months ago
- mkmadata · Added EUR-R13B07 grid support; automated mkmadata · last month
- mkmagrids · Dimension truncation to the land domain · last month
- mksurldata · Tested and improved scripts and documentation (#11) · 2 months ago
- .gitignore · Tested and improved scripts and documentation (#11) · 2 months ago
- LICENSE · add LICENSE · last year
- README.md · ICON is not mentioned anymore to avoid confusion · last month
- jsc.2023_Intel.sh · Tested and improved scripts and documentation (#11) · 2 months ago
- jsc.2024_GCC.sh · Tested and improved scripts and documentation (#11) · 2 months ago
- jsc.2024_Intel.sh · Tested and improved scripts and documentation (#11) · 2 months ago
- mod_domain.sh · Tested and improved scripts and documentation (#11) · 2 months ago

About: This project provides an independent workflow to create surface files for eCLM (CLM5). Originally tailored to curvilinear grids, the workflow is suited for any type of grid.

Releases: No releases published. Create a new release.

Packages: No packages published. Publish your first package.

Contributors: 3

- chartick
- mvhulten Marco van Hulten
- s-poll

eCLM static file generator

This repository shows how to generate curvilinear surface and domain fields for eCLM simulations. The generator follows the official CLM-workflow but makes a few adaptions.



TSMP2 fully coupled simulation experiments, 3km frontier runs

Model configurations, 12km (R13B05, EUR-11), 3km (R13B07, EUR-0275)

- Climate mode configuration
- HWU, LULCC possible
- Non local effects and remote feedbacks with large domain

Grids:

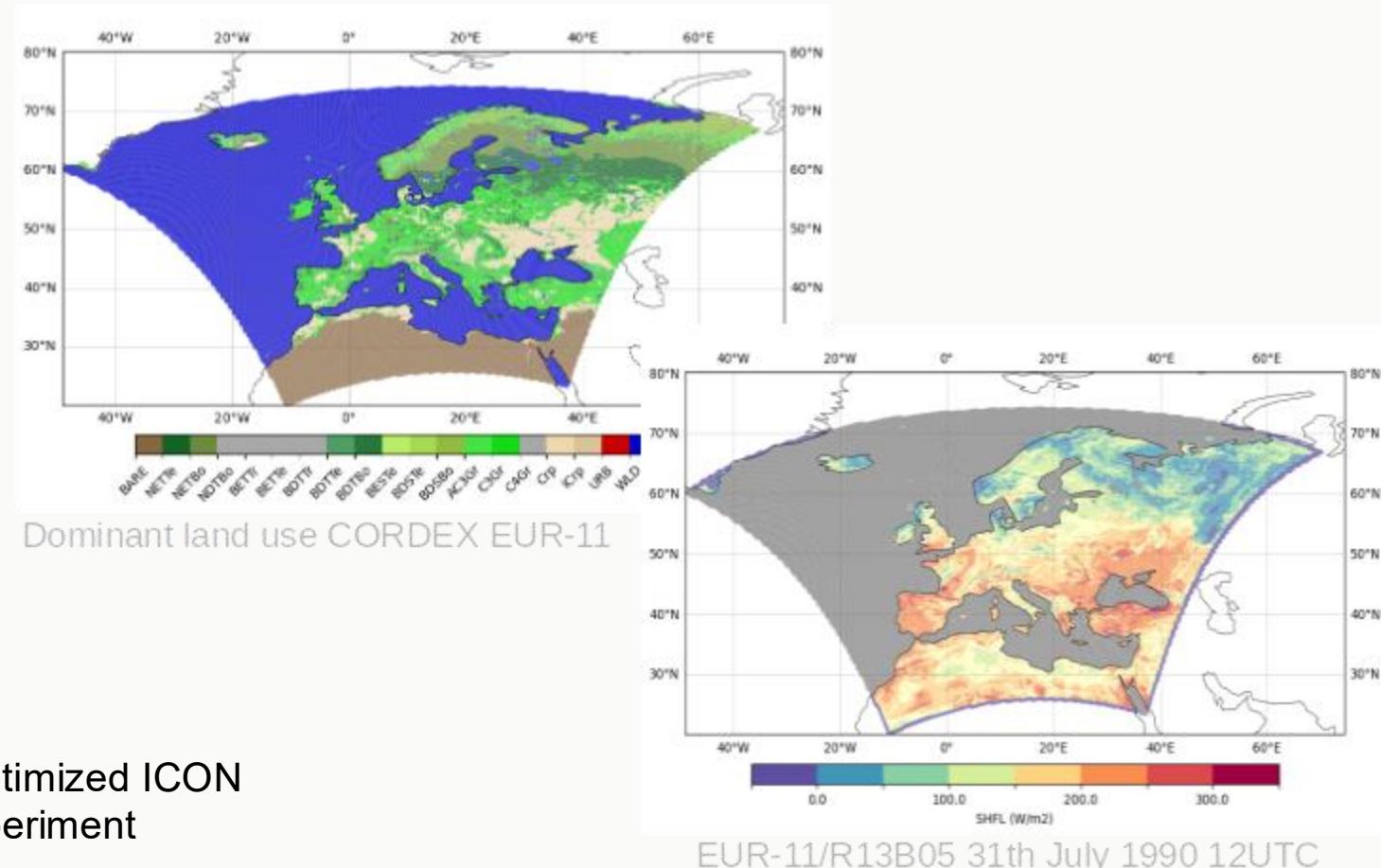
icosahedral (ICON, eCLM)

curvilinear (ParFlow)

4 different model grids:

- 12km domain
 - EUR-11; R13B05
- 3km domain
 - EUR-0275; R13B07

CLMcom, WG CPCS, CECPI-Ger initiative: optimized ICON configuration from coordinated community experiment



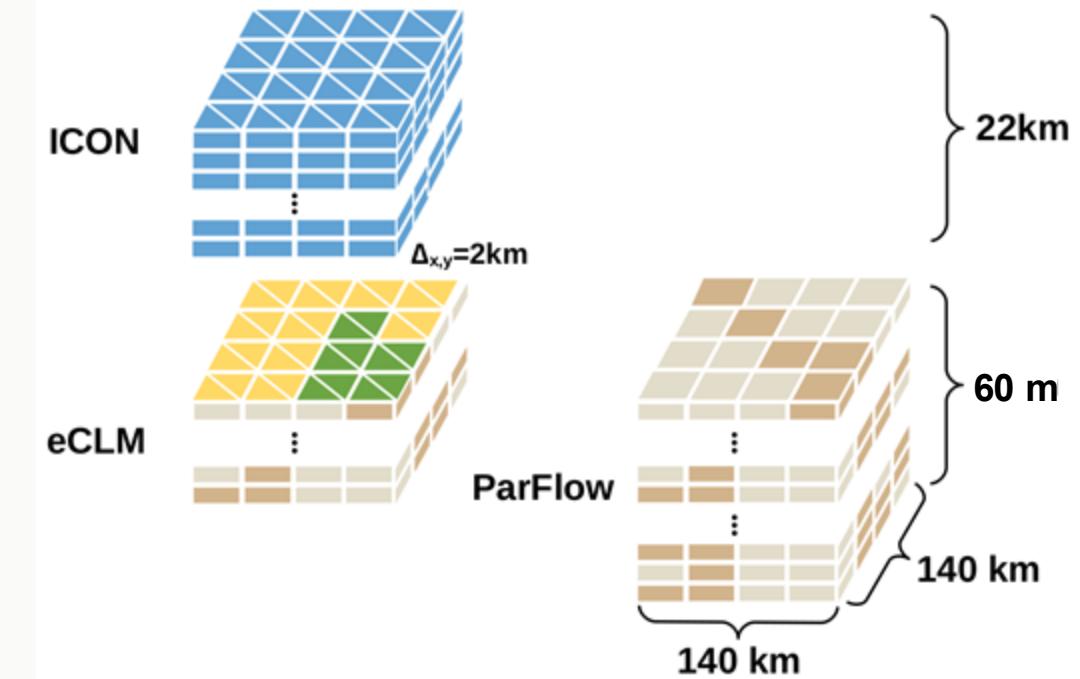


IDEALIZED TEST-CASE

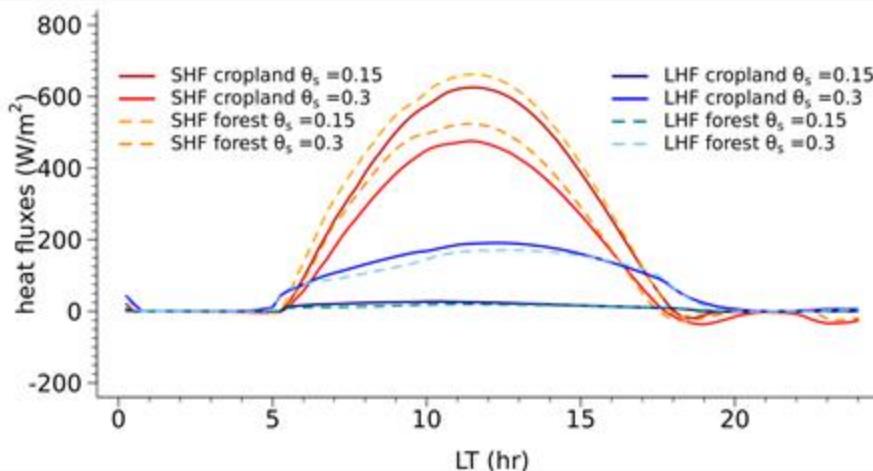
Sensitivity on land-cover and soil moisture

Setup

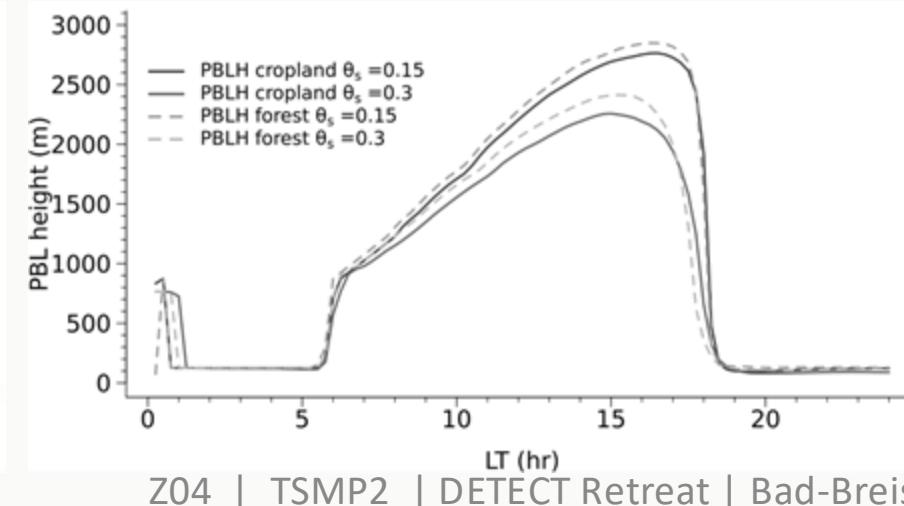
- ICON and eCLM on icosahedral grid,
ParFlow on cartesian grid
- Homogeneous land and sub-surface with
different initial conditions (24hrs)



Feedback on land-cover and soil moisture content:



14/05/2025



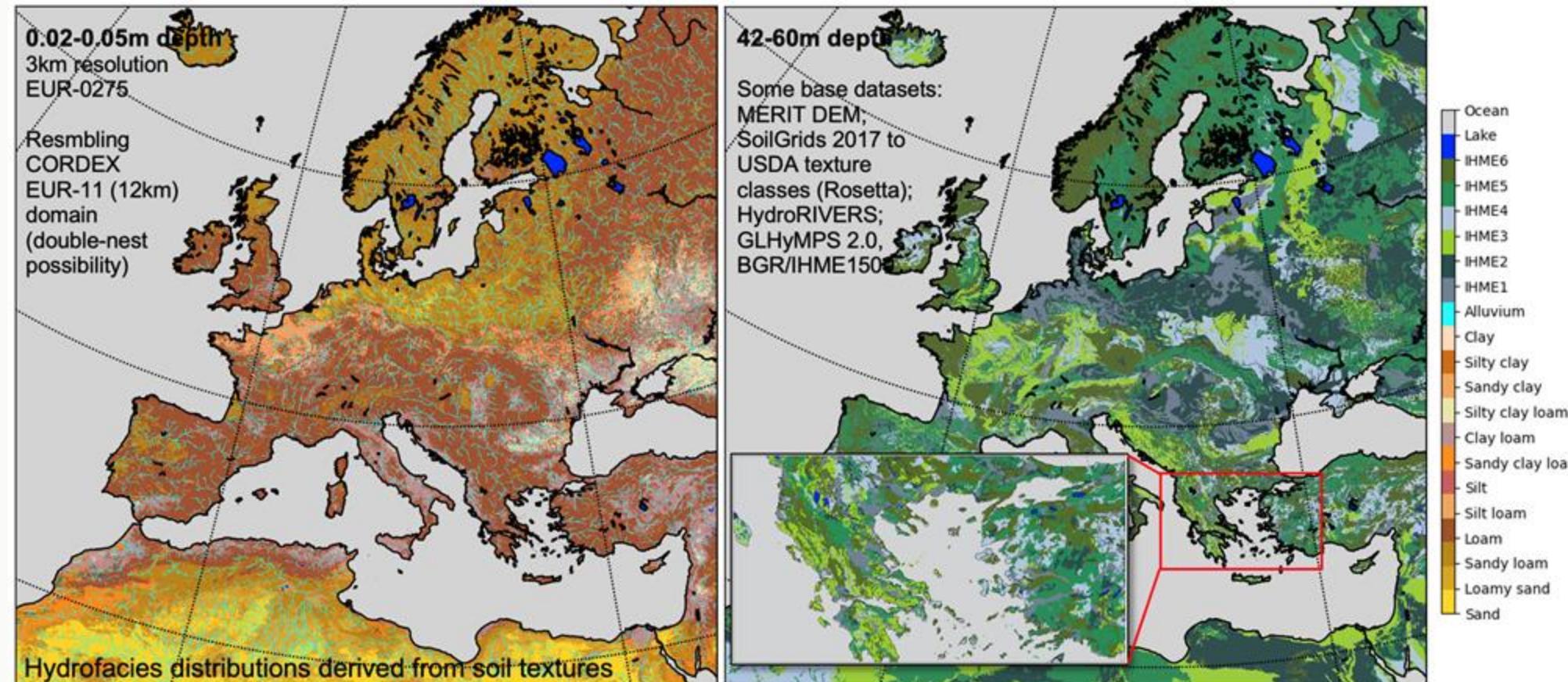
Z04 | TSMP2 | DETECT Retreat | Bad-Breisig

Setup: double periodic boundary condition,
atm. initialisation after
Patton et al. (2005),
loam soil

TSMP2 fully coupled simulation experiments, 3km frontier runs

Model setup and external parameter fields, 12km (R13B05, EUR-11), 3km (R13B07, EUR-0275)

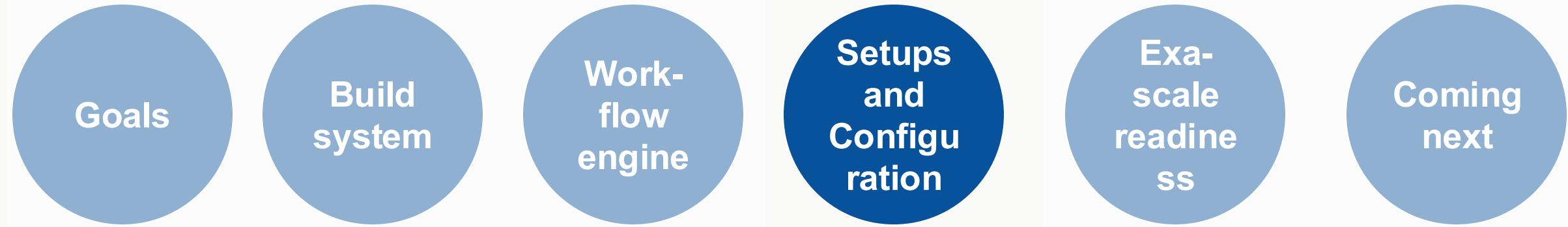
Coupled G2A TSMP2
1600x1552x65/30
3km resolution
Grids: curvilinear
(ParFlow), icosahedral
(ICON, eCLM)
Climate mode
configuration
HWU, LULCC possible
Non local effects and
remote feedbacks with
large domain





CRC1502
DETECT

<https://sfb1502.de/>



Additional tools and functionalities



TSMP2: Water Budget Analysis (WBA) and Data Tools

Goal: Evaluating model physical consistency using scalable data processing tools

Objective 1: Exploring Data Processing Tools

Explore and utilize tools, including Xarray, for handling large-scale model data, particularly focusing on performance and efficiency.

Objective 2: Scaling and Parallelization

Find methods and tools to handle data at different scales in a scalable and parallelizable manner, ensuring efficient processing on high-performance computing (HPC) environments.

Objective 3: Water Budget Calculations, physical consistency

Calculate the water budget for using the tools.

The screenshot shows a GitLab repository interface for 'wba_datatools'. The repository contains several files and a README.md file. The commit history shows multiple commits from 'Kaveh Pataekchi Youseff' over a day ago, all adding ICON visualization files. The README.md file provides details about the repository's purpose and objectives:

TSMP 2.0 Water Budget Analysis - Data tools
Kaveh P. Youseff (BG-3/FZJ) (k.pataekchi.youseff@fz-juelich.de)
This directory contains scripts and tools used for the Water/Energy Budget Analysis of the Terrestrial Systems Modeling Platform (TSMP) 2.0. The main objectives include exploring scalable and parallelizable data processing tools, and calculation of water and energy budgets for the ICON/le-CLM/PerFlow in TSMP 2.0.

Objectives

1. Objective 1: Exploring Data Processing Tools
Explore and utilize tools, including Xarray, for handling large-scale model data, particularly focusing on performance and efficiency.
2. Objective 2: Scaling and Parallelization
Find methods and tools to handle data at different scales in a scalable and parallelizable manner, ensuring efficient processing on high-performance computing (HPC) environments.
3. Objective 3: Water/Energy Budget Calculations
Calculate the water/energy budget for using the tools.

Environment Setup

- `tsmpenv_v1.lsf`
These .lsl files are used to load the necessary HPC environment, modules, and software. See https://tsmp2.scc.kit.edu/organization/hpc_it/lsl_data/novice-and-best-practice/hpc_scientific_operating_environments for more details.

Gitlab repo for WBA and data tools (internal)



TSMP2-PDAF: Data Assimilation with TSMP2

Collaborative development environment

Testing

- <https://icg4geo.icg.kfa-juelich.de/ExternalRepos/tsmp-pdaf/TSMP-PDAF-testcases>
- eCLM-PDAF testcases for SM-DA (10x10, NRW) and Snow-DA (single site)

Python

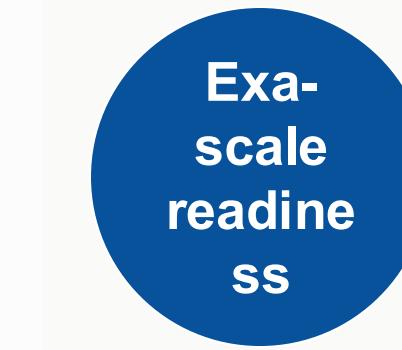
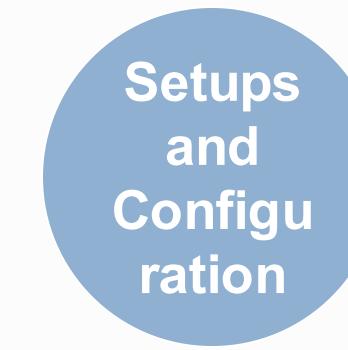
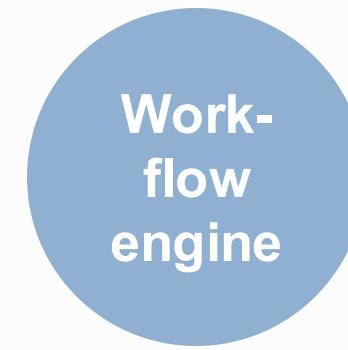
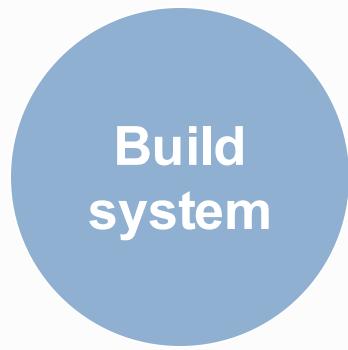
- <https://icg4geo.icg.kfa-juelich.de/ExternalRepos/tsmp-pdaf>
- Observation scripts
- Build scripts
- Preprocessing scripts (perturbation, ensemble namelist)
- Python verification tool

The screenshot shows a GitHub repository named 'TSMP-PDAF'. The repository description states: 'TSMP is a clone of the GitHub repository https://github.com/HPSTerrSys/TSMP. When in doubt, please use the GitHub repository.' The manual can be found at <https://hpscterrsys.github.io/TSMP/>. The repository has several subgroups and projects listed:

- Subgroups and projects: TSMP-PDAF Testcases, TSMP-PDAF Observation Scripts, TSMP-PDAF Build Scripts, TSMP-PDAF Preprocessing Scripts, Python EnKF Verification Tool, TSMP-PDAF Presentations, TSMP-PDAF Component Models, TSMP.
- Shared projects: None.
- Inactive: None.

Search bar: Search (3 character minimum)

File/Folder	Star Rating	Last Updated
TSMP-PDAF Testcases	★ 1	23 hours ago
TSMP-PDAF Observation Scripts	★ 0	5 days ago
TSMP-PDAF Build Scripts	★ 0	5 days ago
TSMP-PDAF Preprocessing Scripts	★ 0	1 week ago
Python EnKF Verification Tool	★ 0	2 weeks ago
TSMP-PDAF Presentations	★ 0	3 weeks ago
TSMP-PDAF Component Models	★ 0	8 months ago
TSMP	★ 1	1 year ago



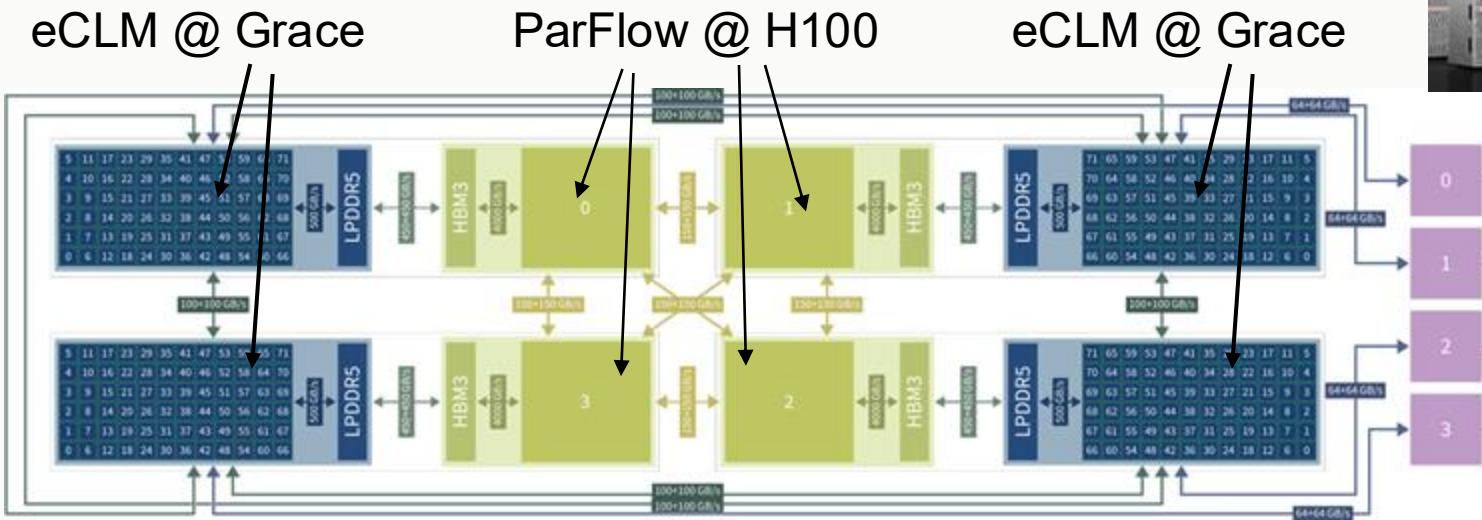
Preparing for exascale

JUPITER will start production soon

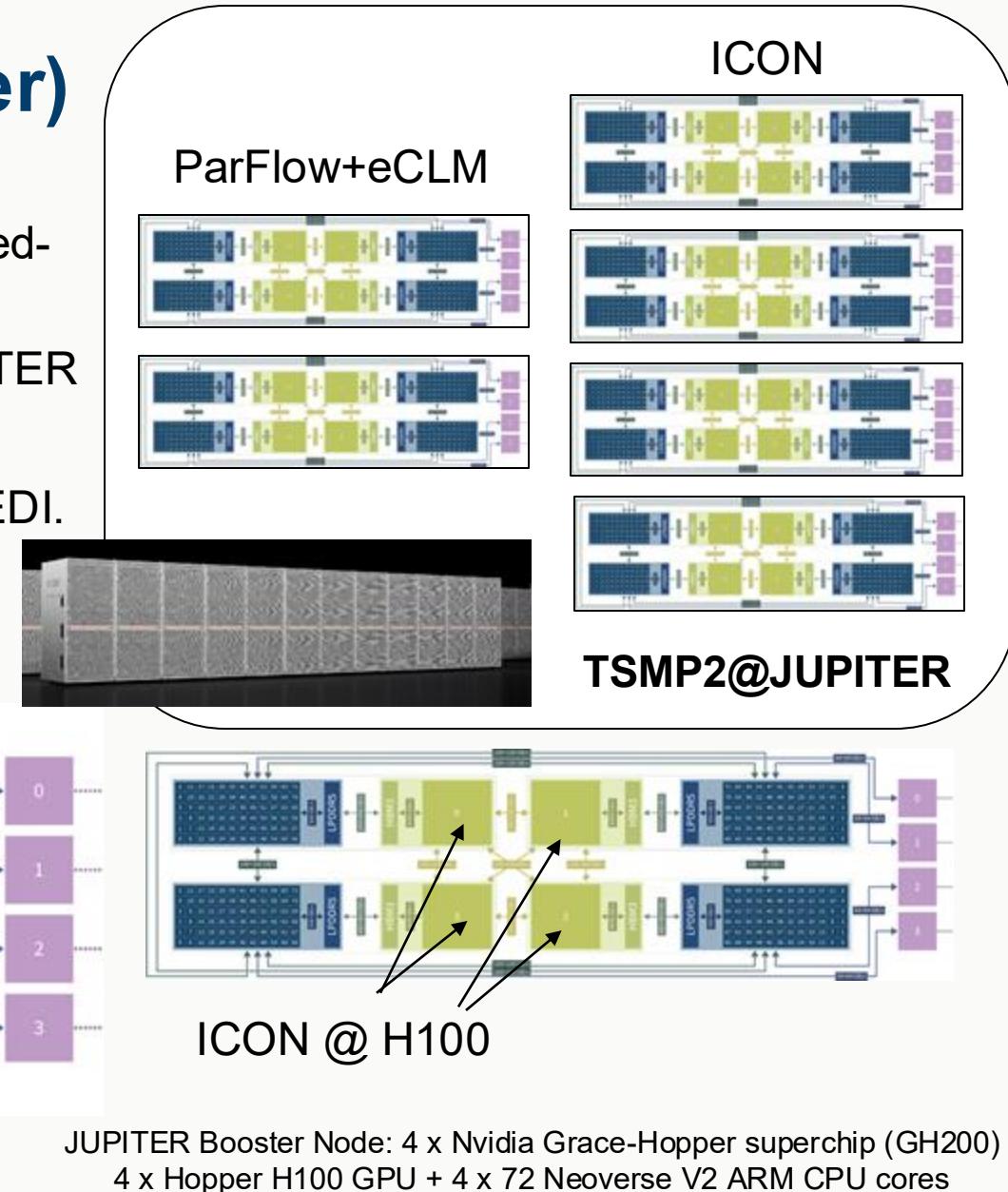


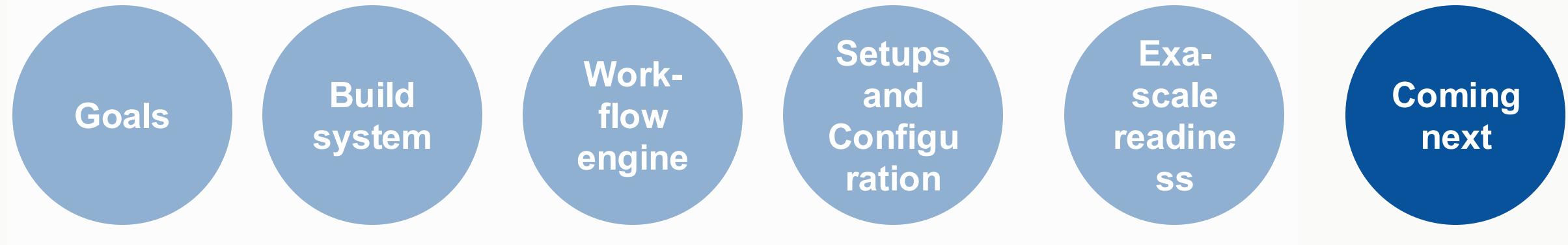
Towards TSMP2 in JUPITER(-Booster)

- ParFlow and ICON on GPUs, CLM on CPU
- Offloading the atmospheric model to GPUs should provide speed-up. Load balancing will completely change
- Exploit in-node heterogeneity in the GH200 superchips in JUPITER
- TSMP2 (CPU) running on JEDI
- First tests with eCLM (CPU) + ParFlow (GPU) are running in JEDI. Adding ICON (GPU) in progress.



In-node heterogeneous computing







CONCLUSION & ONGOING WORK

Terrestrial Systems Modeling Platform (TSMP2) based on ICON, eCLM and ParFlow

Conclusion

- TSMP2 is an open-source, state-of-the-art RESM for comprehensive terrestrial water cycle research G2A
- Computational overhead of ~10-20% due to increase of physical representation
- Software infrastructure for fully coupled system and component models

Ongoing work

- Preparation of baseline simulation 2000-2020 for $\Delta_x=12\text{km}$ and $\Delta_x=3\text{km}$

TSMP2

<https://github.com/HPSCTerrSys/TSMP2>

Component models

ICON

<https://gitlab.dkrz.de/icon/icon-model>

eCLM

<https://github.com/HPSCTerrSys/eCLM>

ParFlow

<https://github.com/parflow/parflow>



Next steps

- Finish ongoing tests, tuning
- “TSMP2 introduction” and “TSMP2 frontier experiment” papers
- Start ERA5 reanalysis-driven 3km evaluation runs asap with latest CLMcom / CECPI-Ger configuration
- CI/CB tests for monitoring the system
- With the DETECT modelling community from 2025-07 onwards (help in improving, testing, tuning):
 - Use of build system and workflow engine (for all model combinations)
 - Getting accustomed to coupled TSMP2 in preparation for phase 2
- Make all data available via detectdata, also from short tests



CRC1502
DETECT

<https://sfb1502.de/>



Status of TSMP2 by DETECT/Z04

2025-05-14 | Stefan Poll^{1,2,3}, Marco van Hulten⁴, Johannes Keller^{1,2}, Paul Rigor^{1,2},
Daniel Caviedes-Voullième^{2,3}, Klaus Goergen^{1,2}

Supported by Kaveh Patakchi Yousefi^{1,2}, Niklas Wager^{1,2,*}, Carl Hartick^{1,2,*}, Muhammad Fahad^{1,2},
Ana Gonzalez-Nicolas^{2,3}

¹Institute of Bio- and Geosciences (IBG-3, Agrosphere), Research Centre Jülich (FZJ), Jülich, Germany

²Centre for High-Performance Scientific Computing in Terrestrial Systems (HPSC TerrSys), Geoverbund ABC/J, Jülich, Germany

³Simulation and Data Lab Terrestrial Systems (SDLTS), Jülich Supercomputing Centre (JSC), Research Centre Jülich (FZJ), Jülich, Germany

⁴High-Performance Computing and Analytics Lab (HPC/A-Lab), University of Bonn, Bonn, Germany

*New affiliation