

# ExaJULES: Model development for a JULES LFRic app

Emma Robinson, Rich Ellis, Doug Clark  
UKCEH

Simon Wilson, David Livings, Dave Case, Bryan  
Lawrence, Grenville Lister  
NCAS

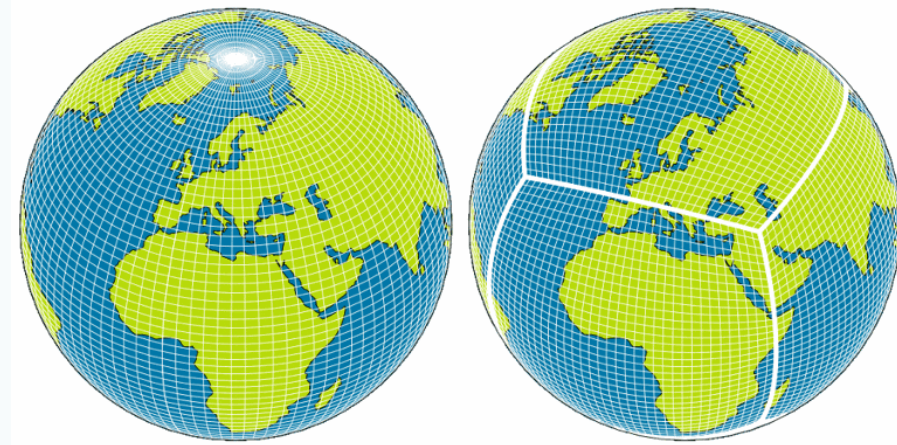
JULES Annual Science Meeting

5<sup>th</sup> September 2024



# LFRic and Momentum

- LFRic software infrastructure
  - Take advantage of next generation exascale platforms
  - Momentum to replace UM
  - GungHo: New dynamical core
  - Psyclone: Auto-generation of parallel code
  - Xios: New approach to i/o
- Forecasts running in parallel with UM
- ExCALIBUR SPF Weather & Climate Use Case supporting redesign of component model codes



**Separation of concerns**  
Science code should be  
agnostic to how it's being  
called

# JULES in LFRic

- JULES code is 'LFRic ready' in as much as it can be compiled as part of atmosphere BUT
  - Not good 'separation of concerns'
  - Different implementation for standalone vs atmosphere
  - JULES standalone can't take advantage of the LFRic technical advances
- LFRic technical infrastructure designed around 'core' and 'apps'

ExaJULES project will design a prototype JULES LFRic app

# ExaJULES

Produce prototype app

- Benchmarking
- Performance improvements
- Coupling components on varying grids
- Engage with JULES and ExCALIBUR communities
- Apr 2023 – Oct 2024



UK Centre for  
Ecology & Hydrology

- Emma Robinson
- Rich Ellis
- Doug Clark



National Centre for  
Atmospheric Science  
NATURAL ENVIRONMENT RESEARCH COUNCIL

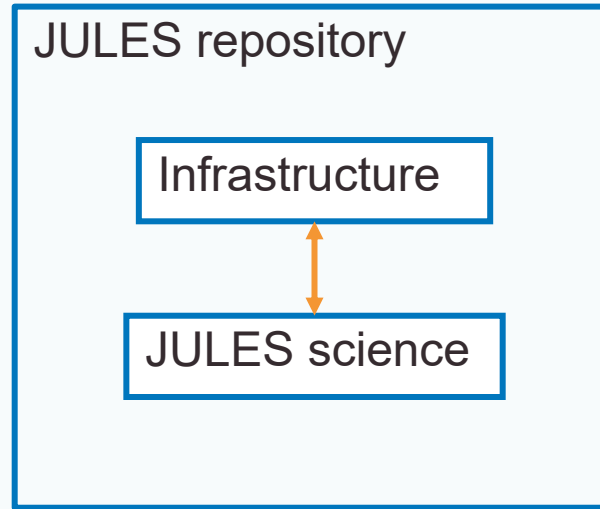
- Bryan Lawrence
- Grenville Lister
- Simon Wilson
- Dave Case
- David Livings



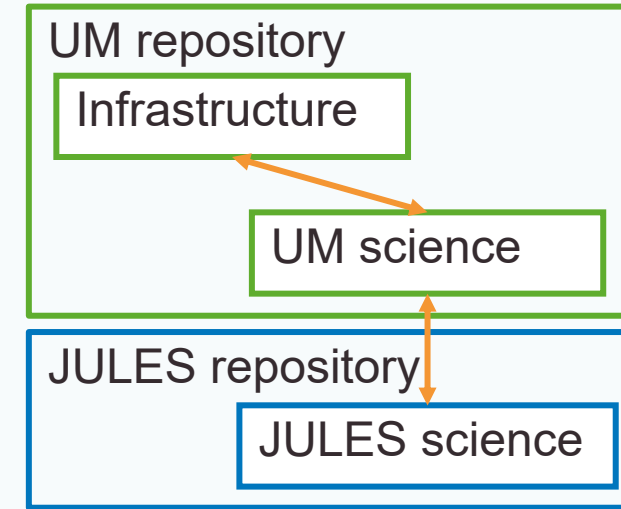
- Giorgia Line
- Rich Gilham
- Martin Best
- Nigel Wood

# Current JULES

## JULES standalone

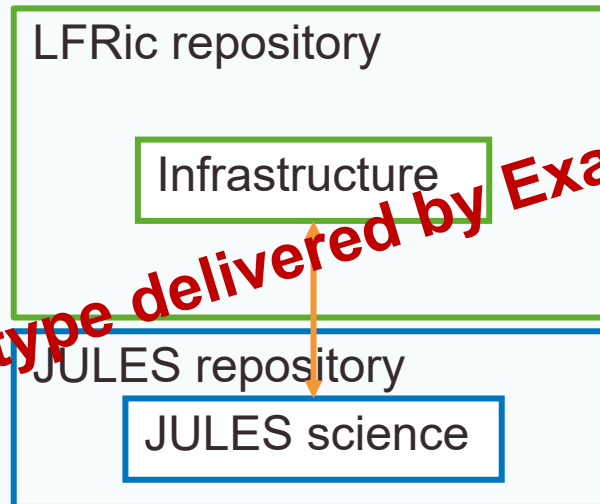


## Coupled to atmosphere



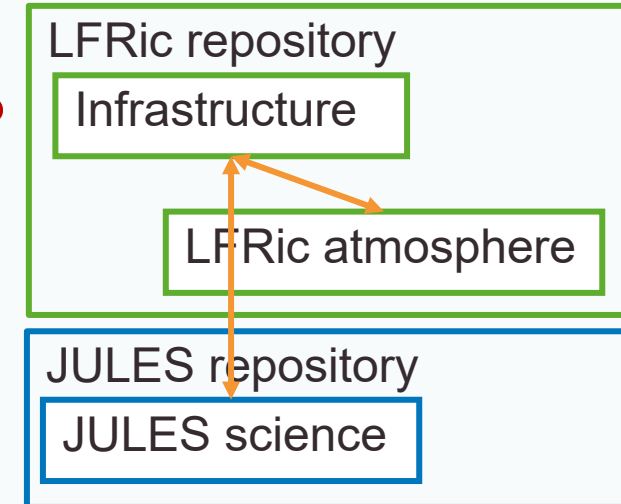
# LFRic JULES

## LFRic JULES standalone



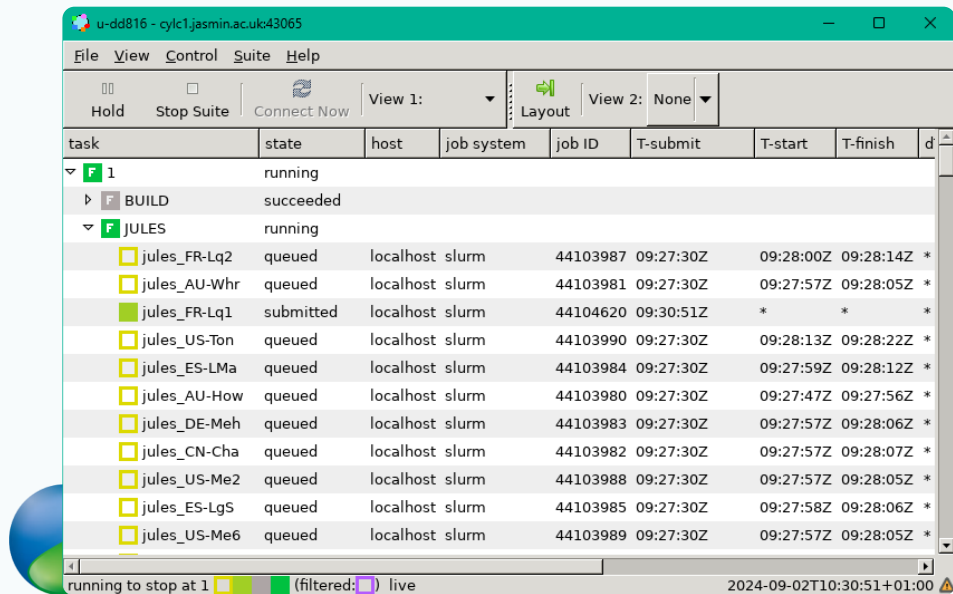
**Prototype delivered by ExaJULES**

## LFRic coupled to atmosphere



# Benchmarking

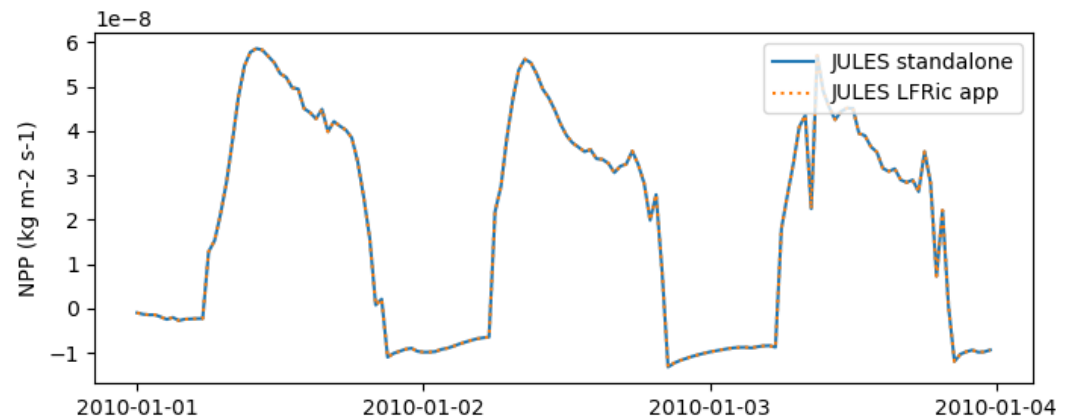
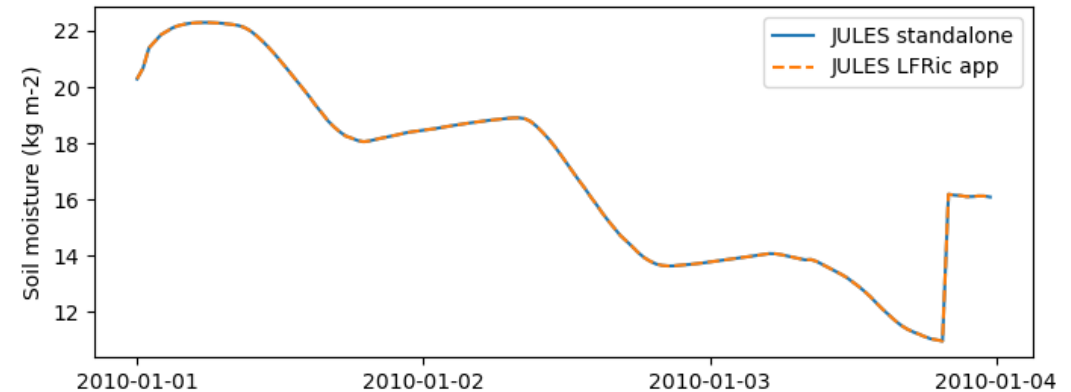
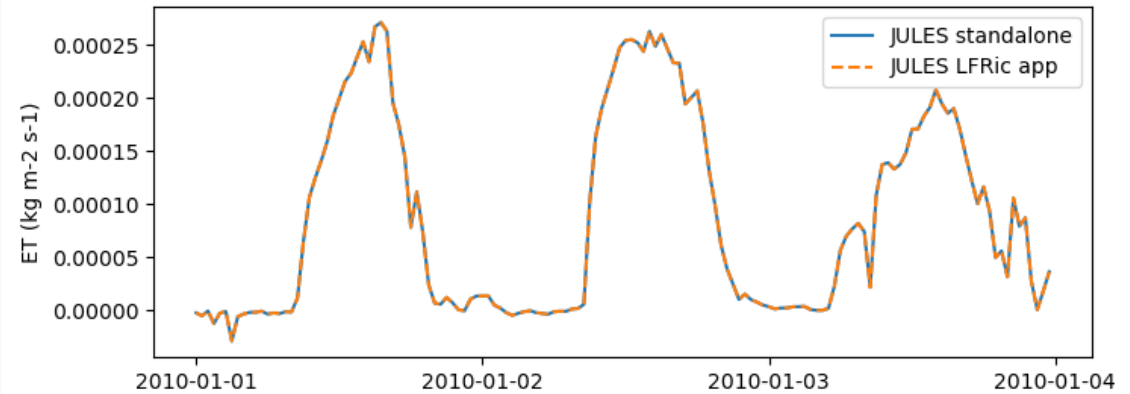
- Adapted JULES-PL benchmarking suite to run with new app
- Testing with selected sites
- Runs on JASMIN with singularity container
- Test runs bit compare with JULES standalone



Terminal window showing task execution details for JULES benchmarking. The window title is "u-dd816 - cylc1.jasmin.ac.uk43065". The interface includes a menu bar (File, View, Control, Suite, Help) and a toolbar with buttons like "Hold", "Stop Suite", and "Connect Now". A table lists the tasks and their status.

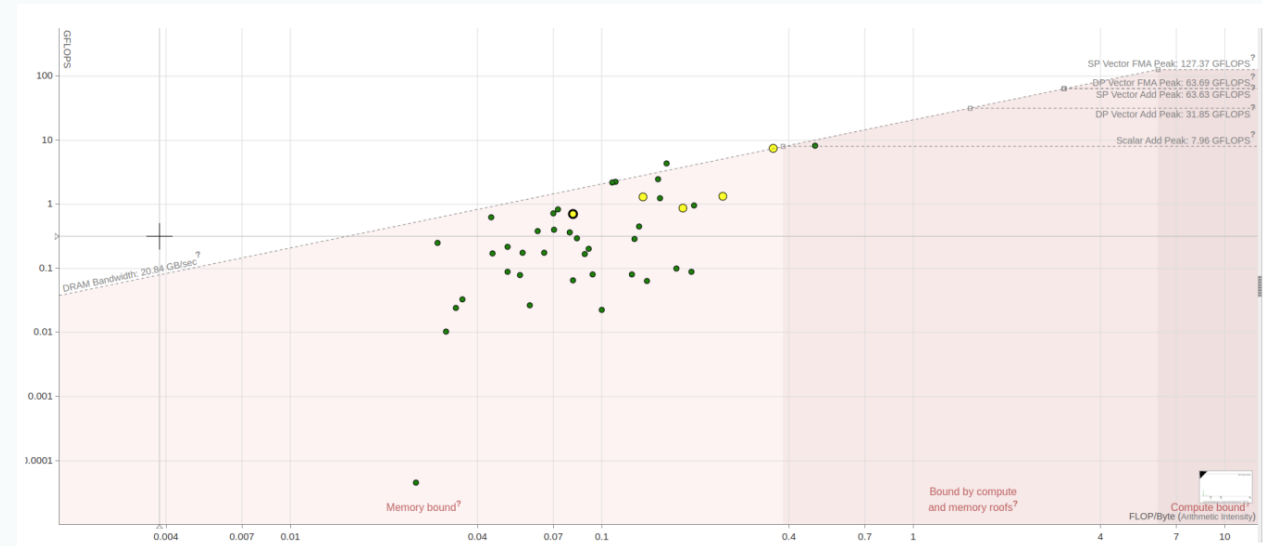
task	state	host	job system	job ID	T-submit	T-start	T-finish	d
1	running							
BUILD	succeeded							
JULES	running							
jules_FR-Lq2	queued	localhost	slurm	44103987	09:27:30Z	09:28:00Z	09:28:14Z	*
jules_AU-Whr	queued	localhost	slurm	44103981	09:27:30Z	09:27:57Z	09:28:05Z	*
jules_FR-Lq1	submitted	localhost	slurm	44104620	09:30:51Z	*	*	*
jules_US-Ton	queued	localhost	slurm	44103990	09:27:30Z	09:28:13Z	09:28:22Z	*
jules_ES-LMa	queued	localhost	slurm	44103984	09:27:30Z	09:27:59Z	09:28:12Z	*
jules_AU-How	queued	localhost	slurm	44103980	09:27:30Z	09:27:47Z	09:27:56Z	*
jules_DE-Meh	queued	localhost	slurm	44103983	09:27:30Z	09:27:57Z	09:28:06Z	*
jules_CN-Cha	queued	localhost	slurm	44103982	09:27:30Z	09:27:57Z	09:28:07Z	*
jules_US-Me2	queued	localhost	slurm	44103988	09:27:30Z	09:27:57Z	09:28:05Z	*
jules_ES-Lg5	queued	localhost	slurm	44103985	09:27:30Z	09:27:58Z	09:28:06Z	*
jules_US-Me6	queued	localhost	slurm	44103989	09:27:30Z	09:27:57Z	09:28:05Z	*

## AR-SLu

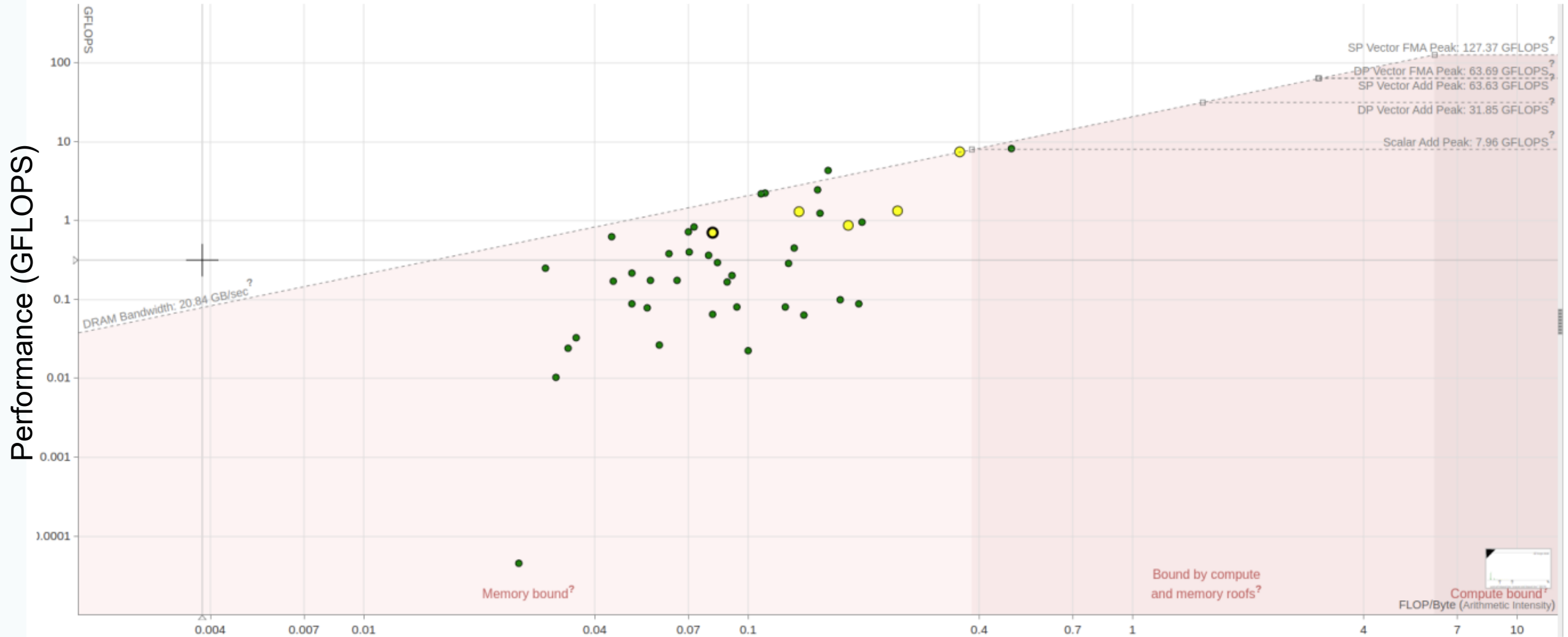


# Performance and optimisation

- Aim: identify and seize optimisation opportunities
- JULES is embarrassingly parallel (except rivers)
- Inherits UM parallelisation strategy (MPI, land vector)
- Standalone is I/O limited
- Use Intel Advisor tool on Archer and JASMIN to identify possible code improvements



# Roofline plot for loops in JULES





# Outcomes

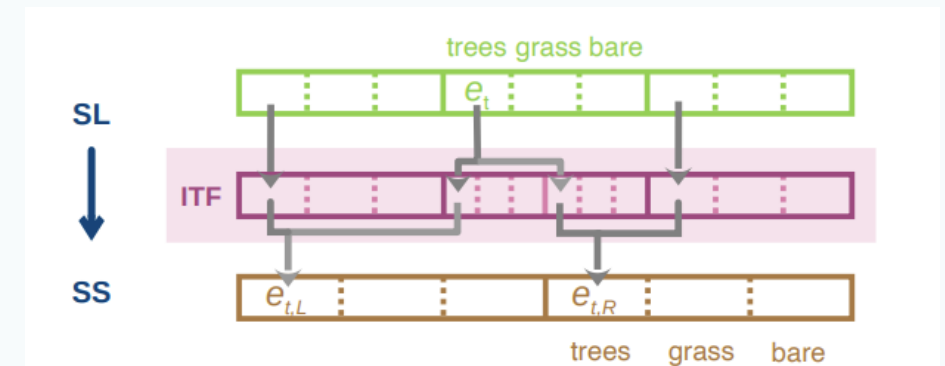
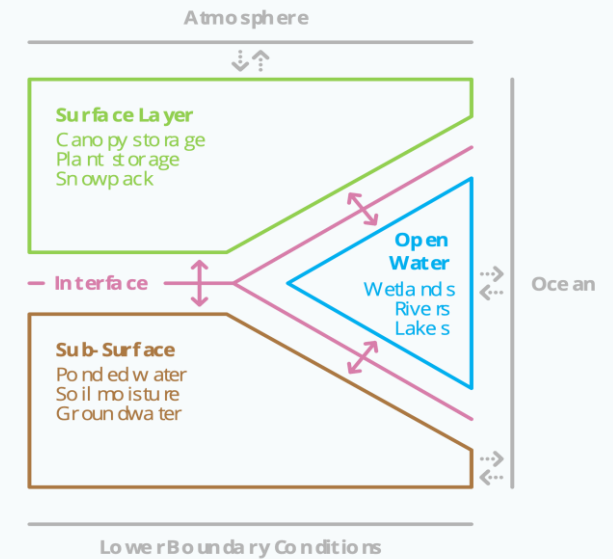
- Identified code changes to improve performance
  - `calc_fsat` worst performance – code changes can lead to 20-25% improvement
  - `root_frac_jls` less important overall – bigger relative improvement of 60%
- More scope for improvement by changing compiler optimisation flags
  - But these will change results
- Have identified possible changes to compiler flags on JASMIN (~30% speed up)
- Will investigate the upcoming changes on JASMIN (new OS, new intel compiler)
- Should continuously monitor performance with code/optimisation changes
- New project NG-ARCH will be investigating parallelisation for W&C codes

# Coupling on multiple meshes

- Investigate options for coupling land to atmosphere on different meshes
- Informed by Hydro-JULES / UnifHy work

## Recommendations

- Task-based parallelisation
- Interaction of tiles and gridboxes
- Need exchange grid / supermesh to handle non-linear processes
- Combine with formal coupling



# Prototype JULES app

## Potential benefits:

- Shared technical infrastructure
- Simplified pull through of science from JULES to coupled model
- Allows JULES to exploit developments in supercomputer infrastructure
- Potential performance gains

## Potential concerns:

- Technical overhead for running standalone
- Does it simplify development?
- Coupling and parallelisation

# Next steps

- ExaJULES...
  - Merge changes to latest LFRic/JULES versions
  - Create tickets to get infrastructure into trunk
- ...and beyond
  - Work towards consistent calling code in standalone and atmosphere
  - Finish interpolation and other outstanding issues
  - Consider different options for coupling to atmosphere
  - Liaise with JULES standalone users

# ExaJULES

- Early prototyping of standalone JULES in LFRic
- Exploring options for future development
- No immediate change to JULES working practices
- Any future change to be supported with required training

# Thank You

For more information  
please contact:  
[emrobi@ceh.ac.uk](mailto:emrobi@ceh.ac.uk)



UK Centre for  
Ecology & Hydrology

