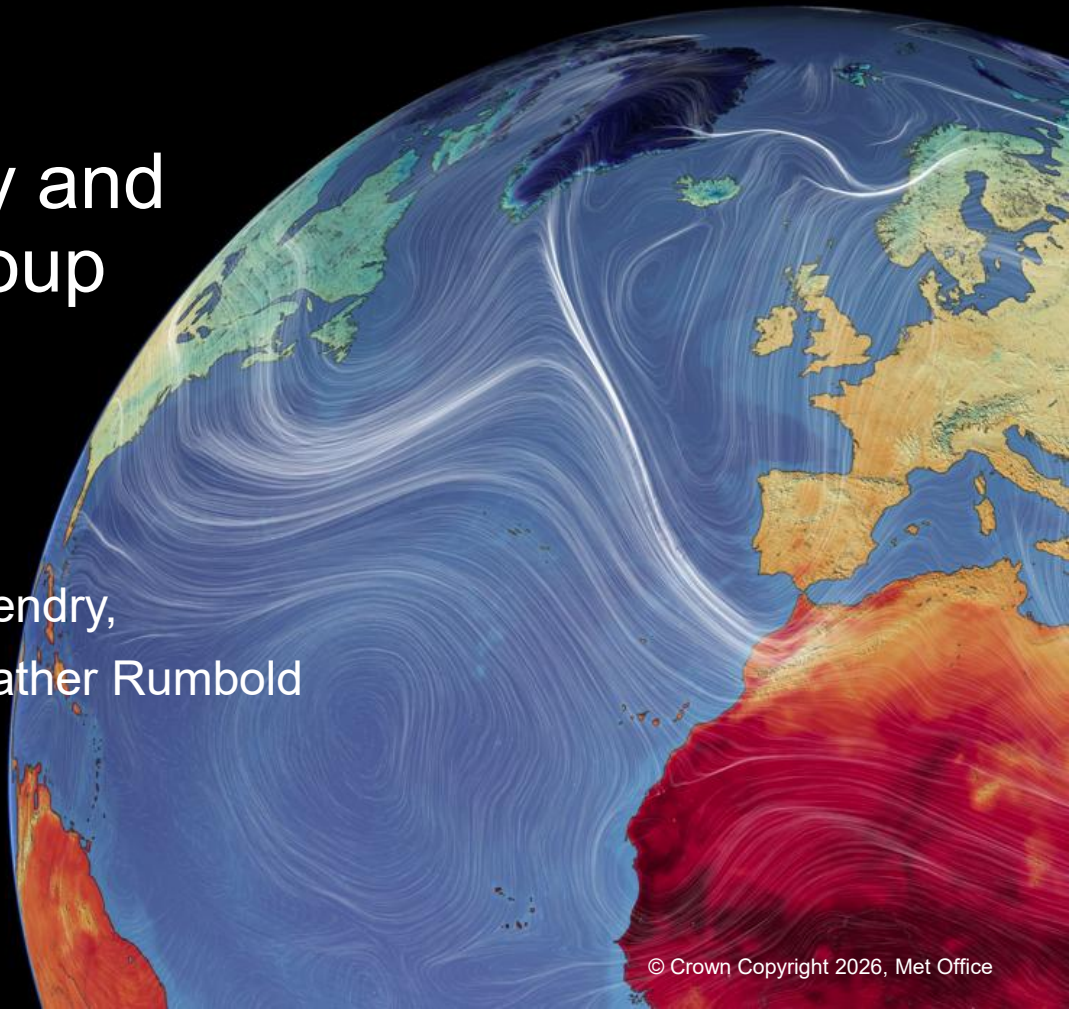


# Update from Hydrology and Surface Processes Group

Gabriel Rooney,  
Martin Best, Joshua Dale, Maggie Hendry,  
Helen Johnson, Simon Osborne, Heather Rumbold  
September 2026

This content has been created by the author.  
AI [M365 Copilot] has also been applied to refine the document.

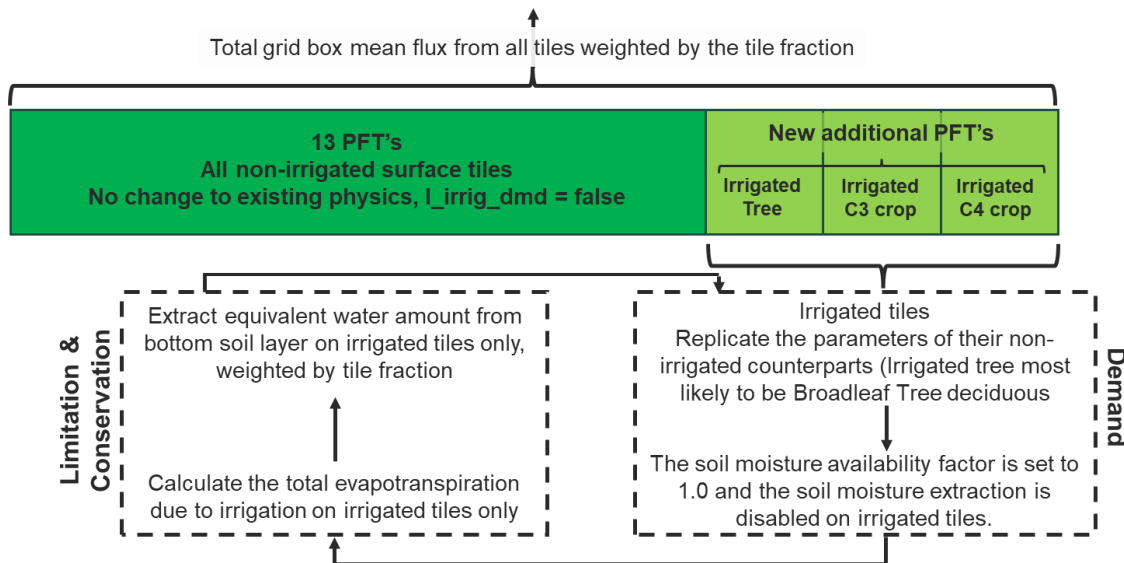


# contents

- irrigation
- Leaf-area index (LAI)
- FLake lake scheme
- JULES metadata
- JULES public on github



# Development of the new tile-based irrigation scheme

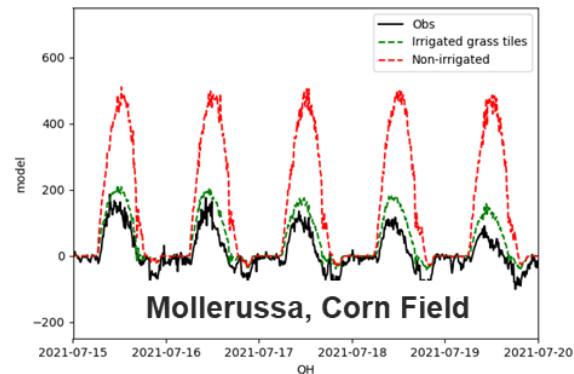


Developed a new **tile-based irrigation scheme** for JULES which **meets future requirements**.

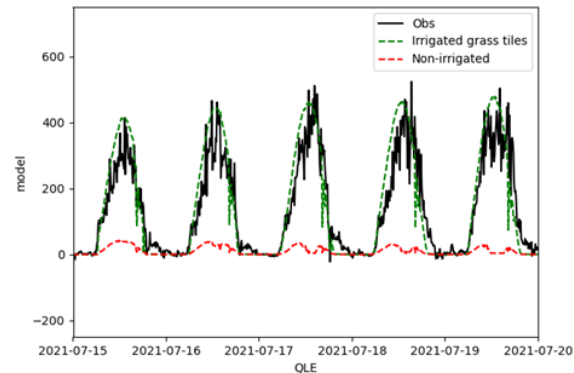
Priorities:

- **Global and regional** testing of the scheme implemented in LFRic
- Implementation of **water conservation** and **water extraction**.
- Developing code for **irrigation ancillaries** within **ANTS**.
- Scoping the practicalities for integrating with the existing **JULES soil tiling** code.

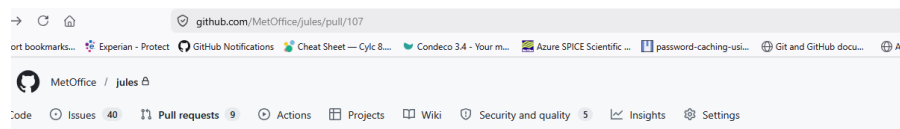
Sensible Heat Flux @3m



Latent Heat Flux @3m



- Technical code to add **two new surface tiles** - C3\_irrig and C4\_irrig
- The science code required to calculate irrigation on the new surface tiles. The  $\beta$  is set to 1.0 and the **soil moisture extraction is disabled** on irrigated tiles.



Implement new irrigation scheme into standalone JULES #107

Open

Heather Rumbold (hsrcumbold) wants to merge 60 commits into MetOffice:main from hsrcumbold:new\_irrigated\_branch\_tile\_irri

Conversation 80

Commits 60

Checks 7

Files changed 44



Heather Rumbold (hsrcumbold) commented on Apr 23 • edited by james-bruten-mo

## PR Summary

Sci/Tech Reviewer: Maggie (@maggiehendry)

Code Reviewer: James Bruten (@james-bruten-mo)

The existing fraction-based irrigation scheme separates the grid box into non-irrigated and irrigated fractions with separate prognostic soil moistures (and therefore separate soil columns) calculated for both. This scheme can split irrigated fractions across different vegetated surface tiles but not in a very flexible clean way. In order to meet the requirements, set out for future developments, it is desirable to write an improved irrigation scheme which will enable us to apply new methods for irrigation on the surface tiles. For this new tile-based scheme, I have created two additional irrigated surface tiles that don't become soil moisture stressed, which replicates the processes that occur during irrigation. All surface tiles in this scheme have a single soil moisture profile and access the same water store, so hence we have prevented irrigated tiles from extracting water from this, so they don't impact their non-irrigated counterparts. In the future, we plan to include a version that combines the best components of both schemes and utilises the currently unused and separate soil tiling code.

- A new variable added to the pftparm namelist to **enable irrigation to be switched on** for individual surface tiles. Plus, associated metadata and upgrade macro required for JULES (and UM).
- A new **irrig\_option switch** has been added to allow this code to be used without impacting the existing irrigation code and allows code to be switched off.
- Standalone **Rose-stem** test to protect the new science added.
- **Technical infrastructure** to allow the code to **couple** to the **LFRic** in a future issue.

# Updates to LAI and canopy height ancillaries

Current Leaf Area Index (LAI) ancillary is based on 5 years of MODIS data at 4km resolution.

Updating the ancillary offers the opportunity to increase the resolution, average over more years of satellite data and make use of improved data processing algorithms.

Two new ancillaries have been produced, one based on MODIS, the other using Copernicus data. Both averaged over 20 years of data.

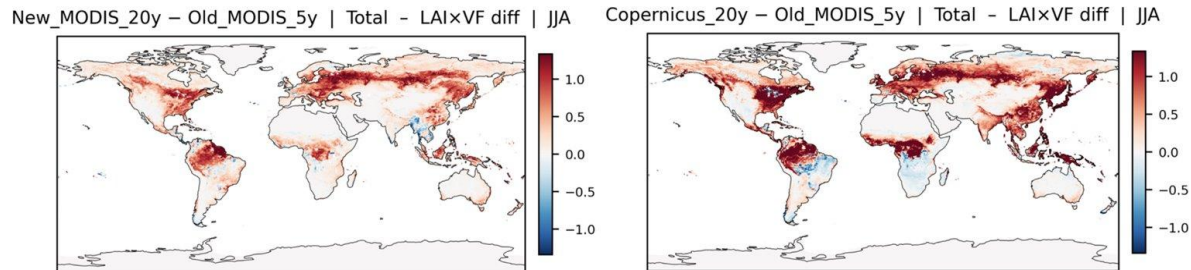


Figure 4—JJA-mean LAI differences relative to the current 4 km ancillary, weighted by vegetation fraction. New MODIS-based ancillary (left) and new Copernicus-based ancillary (right). Red indicates increased LAI; blue indicates decreased LAI.

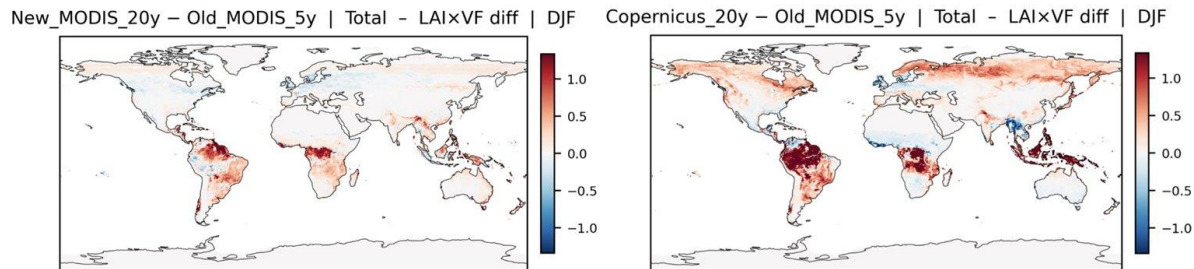
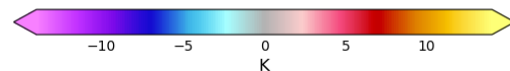
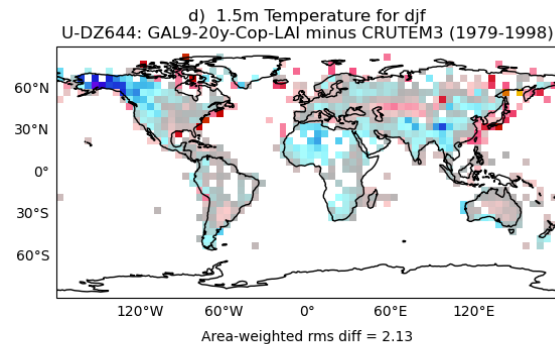
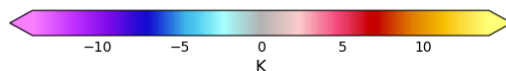
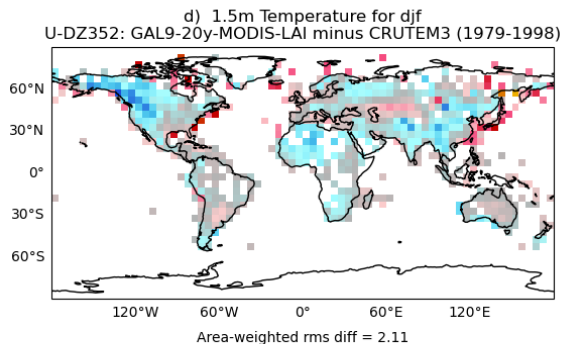
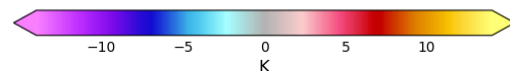
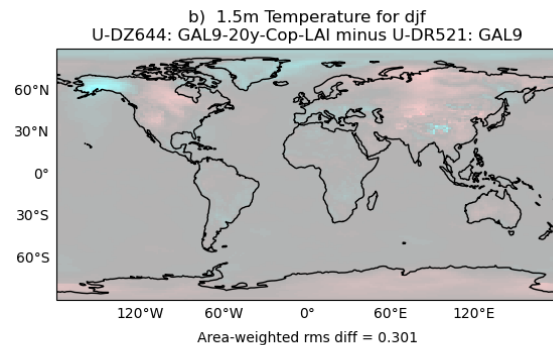
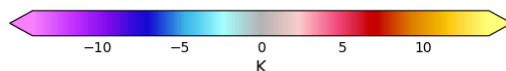
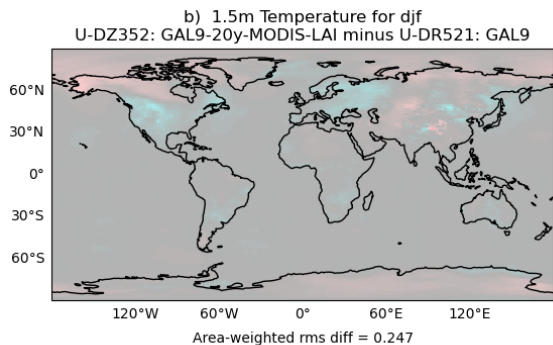


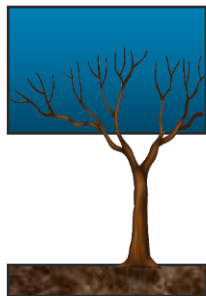
Figure 6—DJF-mean LAI differences relative to the current 4 km ancillary, weighted by vegetation fraction. New MODIS-based ancillary (left) and new Copernicus-based ancillary (right). Red indicates increased LAI; blue indicates decreased LAI.

# Updates to LAI and canopy height ancillaries

The vegetation canopy height ancillary, which is calculated from LAI, has also been updated. In both cases, changes to canopy heights were minimal.

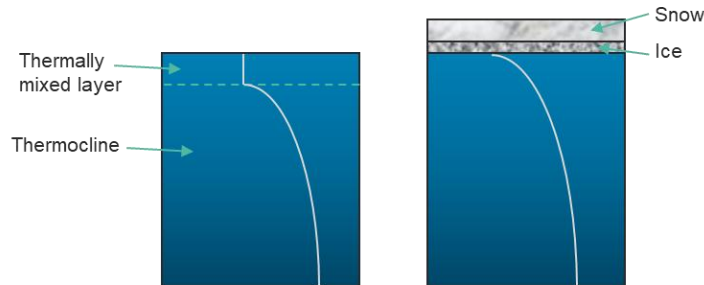
Both new sets of ancillaries have a neutral impact in global climate simulations. This is good, as it will make adopting an updated ancillary straightforward.





Currently, our operational models are using the **vegetation canopy scheme** to represent lakes:

- Specify large **canopy water capacity** and **canopy heat capacity** to mimic a 5m deep lake
- No freezing processes – snow can settle regardless



The Freshwater Lake model '**FLake**' will allow:

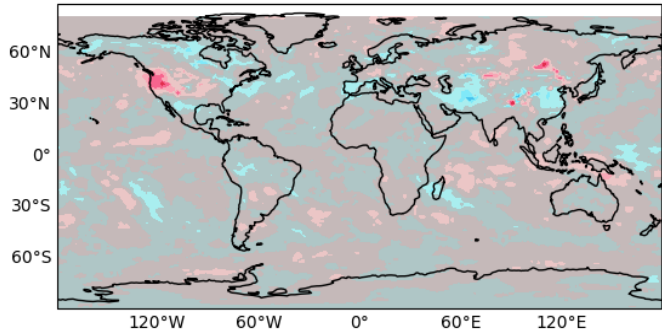
- Different lake depths to be specified for different gridboxes
- Freezing and thawing processes
- Vertical thermal mixing processes

- Currently working to implement FLake in LFRic
- Targeting RAL5 and GC7 configurations

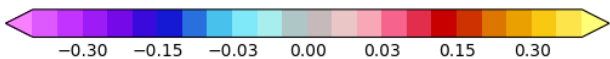
# Improving lake representation with FLake

b) Albedo for djf

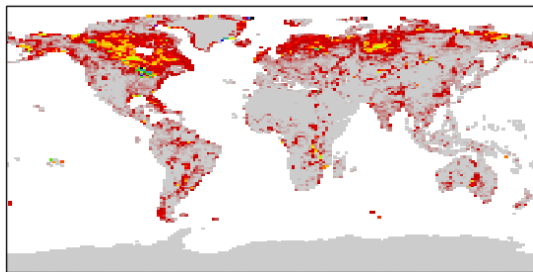
U-CJ297: FLake\_DecoupledFromSoil minus U-CG225: GA8.0



Area-weighted rms diff = 6.49e-03

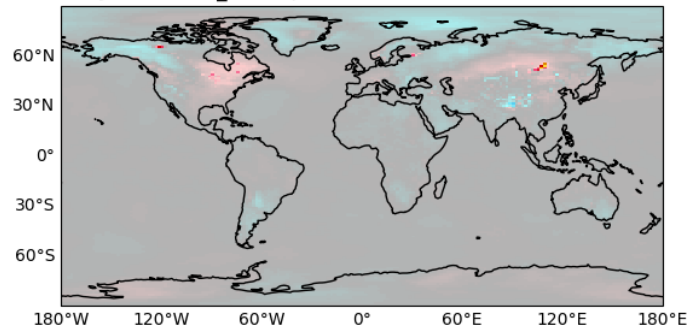


Lake Fraction

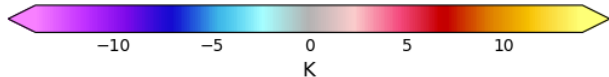


b) 1.5m Temperature for djf

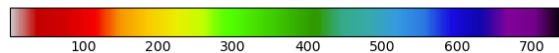
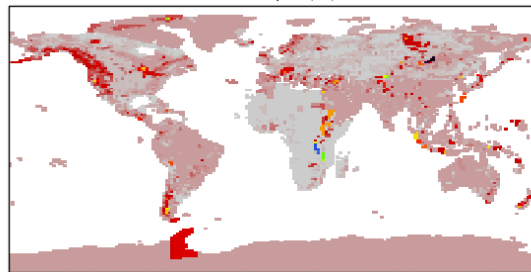
U-CJ297: FLake\_DecoupledFromSoil minus U-CG225: GA8.0



Area-weighted rms diff = 0.337



Lake Depth (m)



# Sharing JULES metadata

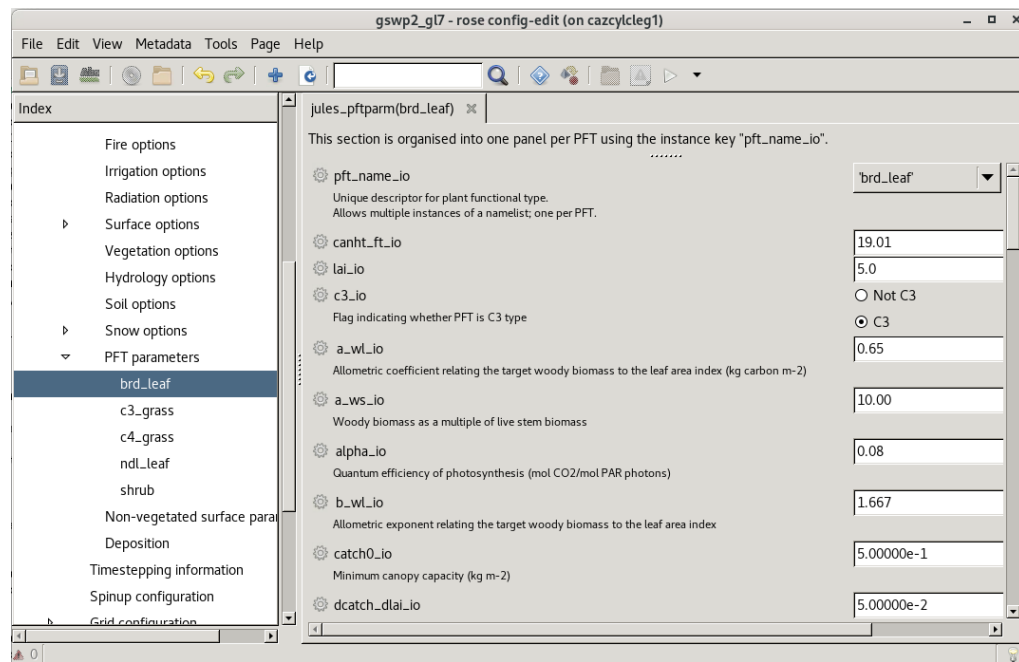
- Migrated and updated the wiki [Sharing JULES metadata](#) to JULES GitHub containing
  - [Structure of metadata](#) description, [What goes where](#), [Making changes to jules-shared](#), [Implications for users & developers](#) including [required LFRic app changes](#)
  - Feedback and suggestions of improvements welcome to maggiehendry via GitHub
- All JULES metadata will reside in the JULES repository
  - From JULES vn8.3 (next release) 🙌
  - [Remove duplicate JULES metadata in UM repository · Issue #154 · MetOffice/jules](#)
  - Some metadata used by LFRic apps still requiring full migration to **jules-shared** (marked “partial” on wiki)
  - jules\_triffid requires consolidation before migration
  - These will be dealt with in separate tickets

# Converting pftparm to one namelist per tile

- From JULES vn8.3 (next release) 🙌
- [Convert `jules\\_pftparm` to a duplicate namelist · Issue #115 · MetOffice/jules](#)
- Namelists with items of size npft, nnvg are challenging to maintain and upgrade with convoluted and incomplete upgrade macros
- Converting to duplicate namelists makes things simpler and more robust
- Will be documented on GitHub wiki
  - [Redesigning JULES tile input · MetOffice/jules Wiki](#)
  - Including Rich Gilham's summary of the issues with the current design from six years ago which this change goes some way to address

# Converting pftparm to one namelist per tile

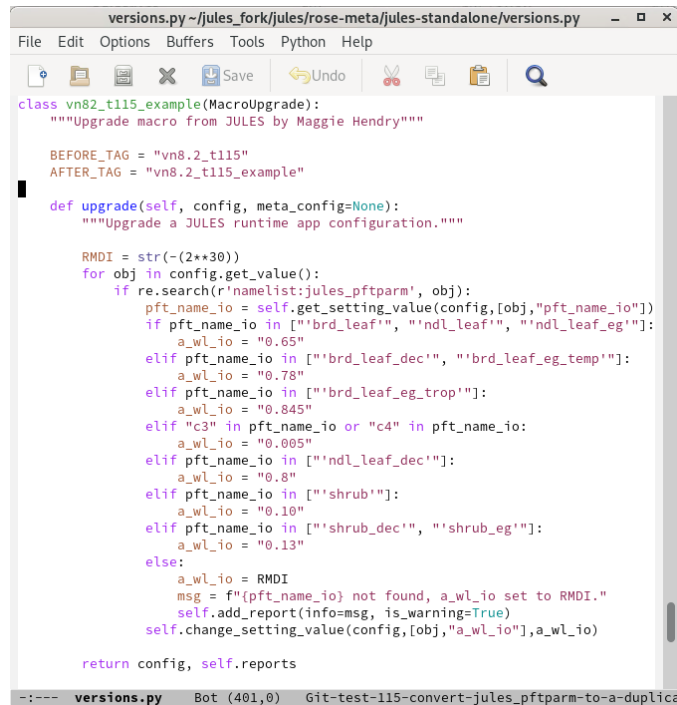
- GUI has one panel per PFT
- Labelled with pft\_name\_io based on jules\_surface\_types



# Converting pftparm to one namelist per tile

- Simpler, more robust upgrade macros
  - Removes the need for a list of npft=5 or npft=9 (ncpft=0 or ncpft=4) or npft=13 or ... which only accounts for known configurations, leading to incomplete macros
  - Does require surface types to be defined in jules\_surface\_types in the code
  - Values are specified by surface type with a default (missing data value) for unspecified types

NB This is an example. I am not changing a\_wl\_io.



```
versions.py ~/jules_fork/jules/rose-meta/jules-standalone/versions.py - □ ×
File Edit Options Buffers Tools Python Help

class vn82_t115_example(MacroUpgrade):
    """Upgrade macro from JULES by Maggie Hendry"""

    BEFORE_TAG = "vn8.2_t115"
    AFTER_TAG = "vn8.2_t115_example"

    def upgrade(self, config, meta_config=None):
        """Upgrade a JULES runtime app configuration."""

        RMDI = str(-(2**30))
        for obj in config.get_value():
            if re.search(r'namelist:jules_pftparm', obj):
                pft_name_io = self.get_setting_value(config, [obj, "pft_name_io"])
                if pft_name_io in ["'brd_leaf'", "'ndl_leaf'", "'ndl_leaf_eg'"]:
                    a_wl_io = "0.65"
                elif pft_name_io in ["'brd_leaf_dec'", "'brd_leaf_eg_temp'"]:
                    a_wl_io = "0.78"
                elif pft_name_io in ["'brd_leaf_eg_trop'"]:
                    a_wl_io = "0.845"
                elif "c3" in pft_name_io or "c4" in pft_name_io:
                    a_wl_io = "0.005"
                elif pft_name_io in ["'ndl_leaf_dec'"]:
                    a_wl_io = "0.8"
                elif pft_name_io in ["'shrub'"]:
                    a_wl_io = "0.10"
                elif pft_name_io in ["'shrub_dec'", "'shrub_eg'"]:
                    a_wl_io = "0.13"
                else:
                    a_wl_io = RMDI
                    msg = f'{pft_name_io} not found, a_wl_io set to RMDI.'
                    self.add_report(info=msg, is_warning=True)
                    self.change_setting_value(config, [obj, "a_wl_io"], a_wl_io)

        return config, self.reports
```

-- versions.py Bot (401,0) Git-test-115-convert-jules\_pftparm-to-a-duplica

# JULES move to public repository on github

- JULES vn8.2 (July 2026)
- The JULES repository is now publicly visible, meaning existing forks will have been disconnected on GitHub.
- Please see [https://metoffice.github.io/simulation-systems/FurtherDetails/visibility\\_changes.html](https://metoffice.github.io/simulation-systems/FurtherDetails/visibility_changes.html) for advice on reattaching a fork.
- Ensure you have setup [verified commits](#) before beginning work on a branch that is targetting being merged to main.