Firedrake

Two more tiny Python packages for scientific computing: mpi-pytest and petsctools

Connor Ward

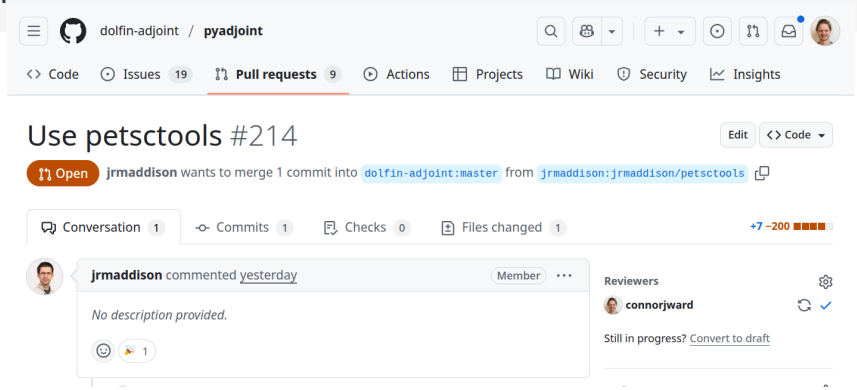
07/07/2025

Imperial College London

About me

- I work on the Firedrake finite element framework
- I write a lot of Python
- I work with HPC + MPI

Motivation




```
130 - # Modified version of flatten_parameters function from firedrake/petsc.py,
131 - # Firedrake master branch 57e21cc8ebdb044c1d8423b48f3dbf70975d5548, first
132 - # added 2024-08-08
133 - def flatten_parameters(parameters, sep="_"):
134 +     """Flatten a nested parameters dict, joining keys with sep.
```


Imperial College London

mpi-pytest

```python
def test_comm_world_size_equals_two():
  assert COMM_WORLD.size == 2


def test_comm_world_size_equals_three():
  assert COMM_WORLD.size == 3
```

```
$ pytest test_comms.py  # won't work!

$ mpiexec -n 2 pytest test_comms.py  # won't work!
```

mpi-pytest

```python
@pytest.mark.parallel(2)
def test_comm_world_size_equals_two():
  assert COMM_WORLD.size == 2

@pytest.mark.parallel(3)
def test_comm_world_size_equals_three():
  assert COMM_WORLD.size == 3
```

```
$ pytest test_comms.py  # works, calls MPI under the hood

$ mpiexec -n 2 pytest test_comms.py -m parallel[2]  # works
```

mpi-pytest

```python
@pytest.mark.parallel(2)
def test_comm_world_size_equals_two():
  assert COMM_WORLD.size == 2
```

```python
@pytest.mark.parallel(3)
def test_comm_world_size_equals_three():
  assert COMM_WORLD.size == 3
```

```
$ pytest test_comms.py  # works, calls MPI under the hood

$ mpiexec -n 2 pytest test_comms.py -m parallel[2]  # works
```

```
pip install mpi-pytest
```

petsctools

- PETSc's Python bindings (petsc4py) mimic the C API
- petsctools provides 'Pythonic extensions'

- PETSc's Python bindings (petsc4py) mimic the C API
- petsctools provides 'Pythonic extensions'

Examples include:

- Managing nested trees of options
- Reading PETSc configuration information
- (TODO) Custom monitors (e.g. plot convergence)
- (TODO) Passing data between Python and PETSc
- And more...

petsctools

- PETSc's Python bindings (petsc4py) mimic the C API
- petsctools provides 'Pythonic extensions'

Examples include:

- Managing nested trees of options
- Reading PETSc configuration information
- (TODO) Custom monitors (e.g. plot convergence)
- (TODO) Passing data between Python and PETSc
- And more...

```
pip install petsctools
```