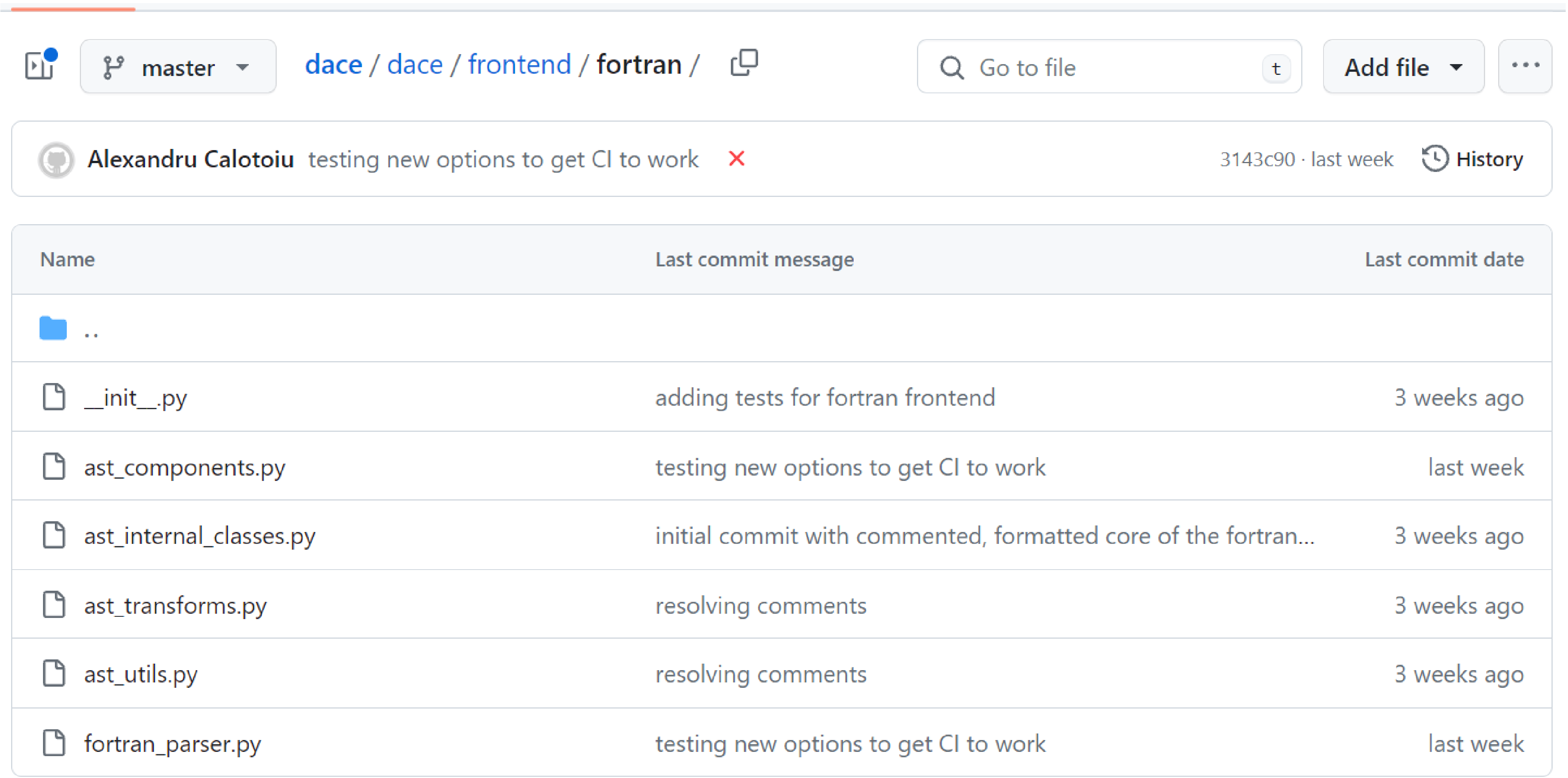
DaFlEx Progress report

Fortran merged!

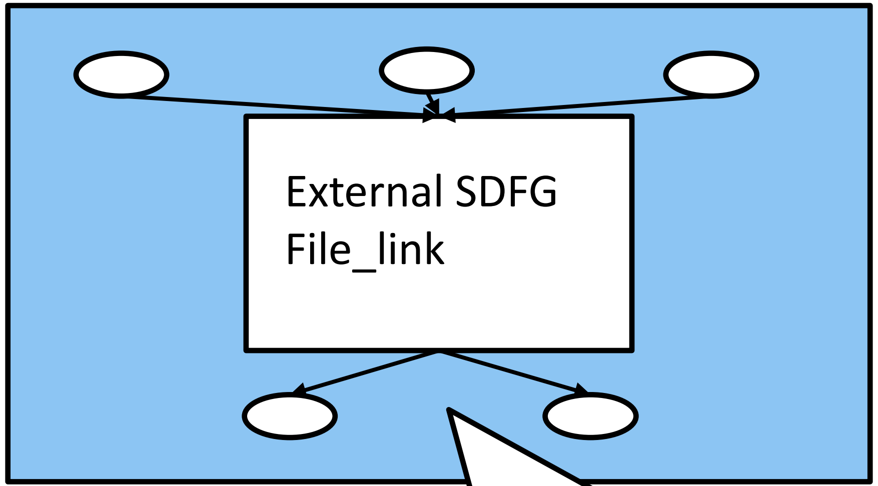


Alexandru Calotoiu testing new options to get CI to work ✗ 3143c90 · last week

Name	Last commit message	Last commit date
..		
__init__.py	adding tests for fortran frontend	3 weeks ago
ast_components.py	testing new options to get CI to work	last week
ast_internal_classes.py	initial commit with commented, formatted core of the fortran...	3 weeks ago
ast_transforms.py	resolving comments	3 weeks ago
ast_utils.py	resolving comments	3 weeks ago
fortran_parser.py	testing new options to get CI to work	last week

External nested SDFGs – In progress

- **Goal – New workflow:**
  - create SDFGs for each function first
  - do local optimization
  - load them together (potentially hierarchically)
  - do global optimization



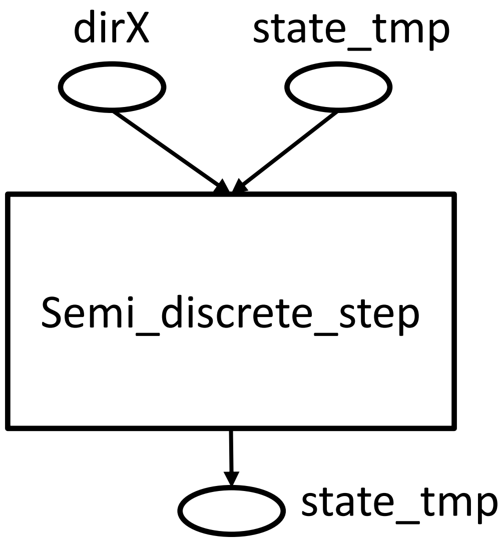
“Slotting” the external SDFG in is not necessarily trivial: What if the NestedSDFG was simplified and no longer uses all arguments?
We can leverage the lessons of the frontends!

```
call semi_discrete_step( state , dirX , state_tmp)
```

Not used

Read

Read &
Written



ICON and Cloverleaf

- Adding more Fortran features
- Increasing robustness

The challenges of large codes

- **Partial compilation**
  - Incomplete allocation information

    Placeholder sizes
  - Incomplete function space

    Placeholder NSDFGs
  - Incomplete structure information

    Partial types
  - Non-compilable SDFG

    Separate process into SDFG per file and multiple SDFG integration