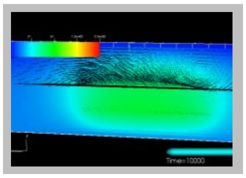
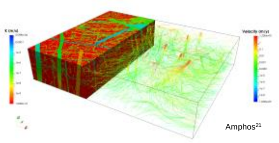
SAND2016-10002PE

Exceptional service in the national interest

Sandia National Laboratories

$$
\begin{aligned}
\frac{\partial m_a}{\partial t} &= -\nabla \cdot \left(\rho_l X_a^l \boldsymbol{q}_l + \rho_g X_a^g \boldsymbol{q}_g + \boldsymbol{J}_a^l + \boldsymbol{J}_a^g\right) + q_a^G,\\
\frac{\partial m_w}{\partial t} &= -\nabla \cdot \left(\rho_l X_w^l \boldsymbol{q}_l + \rho_g X_w^g \boldsymbol{q}_g + \boldsymbol{J}_w^l + \boldsymbol{J}_w^g\right) + q_w^G,\\
\frac{\partial e}{\partial t} &= -\nabla \cdot \left(\rho_l H_l \boldsymbol{q}_l + \rho_g H_g \boldsymbol{q}_g - \kappa_{\text{eff}} \nabla T\right) + q_e^G,
\end{aligned}
$$




Amphos[21]

Introduction to PFLOTRAN

Glenn Hammond

U.S. DEPARTMENT OF ENERGY NNSA

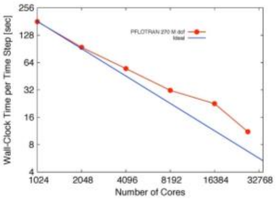
Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

This research is funded by WIPP programs administered by the Office of Environmental Management (EM) of the U.S Department of Energy.

PFLOTRAN

Sandia National Laboratories

- Petascale reactive multiphase flow and transport code
- Open source license (GNU LGPL 2.0)
- Object-oriented Fortran 2003/2008
  - Pointers to procedures
  - Classes (extendable derived types with member procedures)



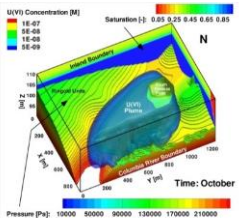
- Founded upon well-known (supported) open source libraries
  - MPI, PETSc, HDF5, METIS/ParMETIS/CMAKE
- Demonstrated performance
  - Maximum # processes: 262,144 (Jaguar supercomputer)
  - Maximum problem size: 3.34 billion degrees of freedom
  - Scales well to over 10K cores

Benefits of Open Source Development

- Encourages collaboration
  - Development
  - Testing
  - Debugging
- Transparency exposes implementation details critical to scientific reproducibility, but excluded by journal publications.
- More optimal use of funding
  - Funding is pooled across a diverse set of projects/budgets.
  - What would have been spend on licensing fees can be redirected towards development.
  - Infinite benefit to those who are unfunded.
- The most fit codes tend to survive (natural selection).
- A community can drive the code to evolve beyond the original vision (evolution).

Application of PFLOTRAN

- Nuclear waste disposal
  - Waste Isolation Pilot Plant (WIPP) in Carlsbad, NM
  - DOE Used Fuel Disposition Program
  - SKB Forsmark Spent Fuel Nuclear Waste Repository (Sweden, Amphos^{21})
- Climate: coupled overland/groundwater flow; CLM
  - Next Generation Ecosystem Experiments (NGEE) Arctic
  - DOE Earth System Modeling (ESM) Program
- Biogeochemical transport modeling
  - U(VI) fate and transport at Hanford 300 Area
  - Hyporheic zone biogeochemical cycling
    - Columbia River, WA, USA
    - East River, CO, USA
- CO_2 sequestration
- Enhanced geothermal energy
- Radioisotope tracers
- Colloid-facilitated transport

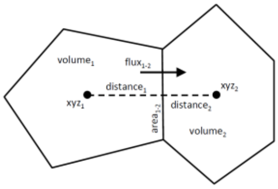


Hammond and Lichtner, WRR, 2010

Numerical Methods

- Spatial discretization
  - Finite volume (2-point flux default)
  - Structured and unstructured grids
- Time discretization: backward Euler
- Nonlinear solver
  - Newton-Raphson
  - Line search/damping with custom convergence criteria
- Linear solver: direct (LU) or iterative (BiCGStab)
- Multi-physics coupling

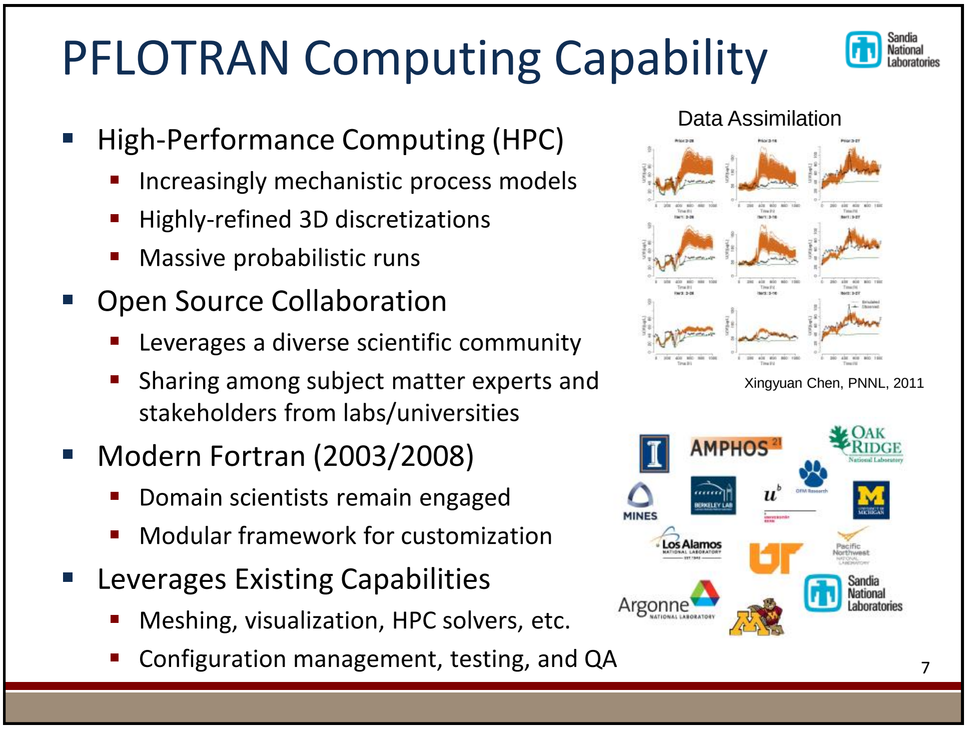
Flow and transport/reaction:	sequential
Transport and reaction:	global implicit
Geomechanics and flow/transport:	sequential
Geophysics and flow/transport:	sequential



PFLOTRAN Multi-Physics Capability

- Flow
  - Single phase, variably-saturated
  - Multiphase gas-liquid
  - Interchangeable constitutive models and equations of state
- Energy
  - Thermal conduction and convection
- Multi-Component Transport
  - Advection
  - Hydrodynamic dispersion
- Chemical Reaction
  - Aqueous speciation
  - Mineral precipitation-dissolution
  - Sorption
  - Microbiological
  - Radioactive decay with daughter products
- Geomechanics
  - Elastic deformation
- Geophysics
  - Coupling to E4D

(Tim Johnson, PNNL)

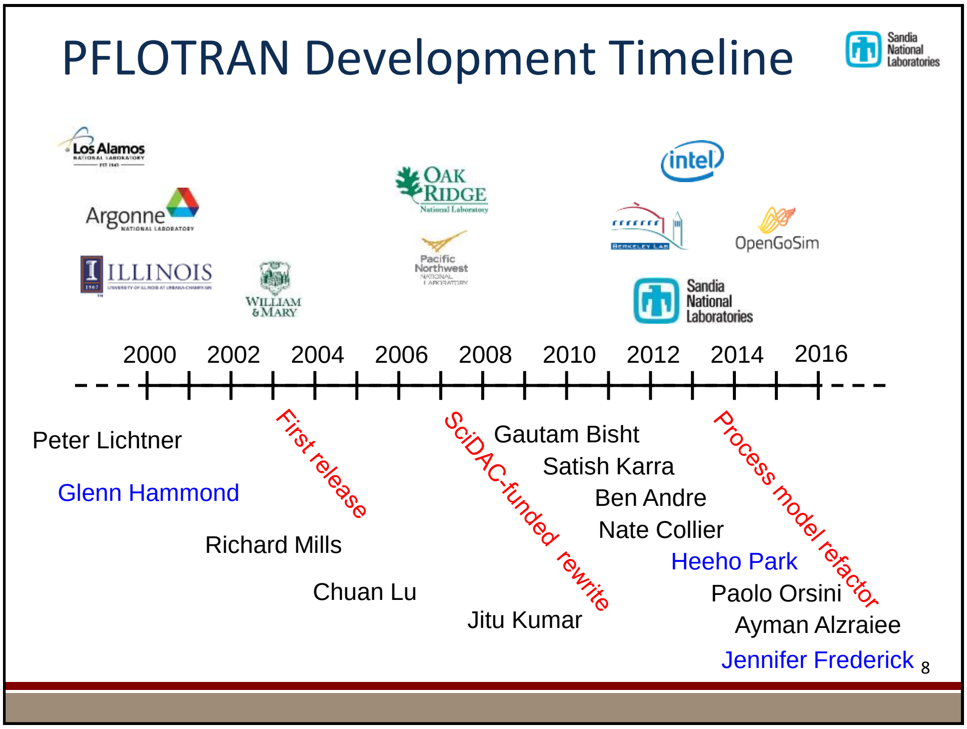


PFLOTRAN Computing Capability

- High-Performance Computing (HPC)
  - Increasingly mechanistic process models
  - Highly-refined 3D discretizations
  - Massive probabilistic runs
- Open Source Collaboration
  - Leverages a diverse scientific community
  - Sharing among subject matter experts and stakeholders from labs/universities
- Modern Fortran (2003/2008)
  - Domain scientists remain engaged
  - Modular framework for customization
- Leverages Existing Capabilities
  - Meshing, visualization, HPC solvers, etc.
  - Configuration management, testing, and QA

Data Assimilation

Xingyuan Chen, PNNL, 2011



PFLOTRAN Development Timeline

Peter Lichtner

Glenn Hammond

Richard Mills

Chuan Lu

Gautam Bisht

Satish Karra

Ben Andre

Nate Collier

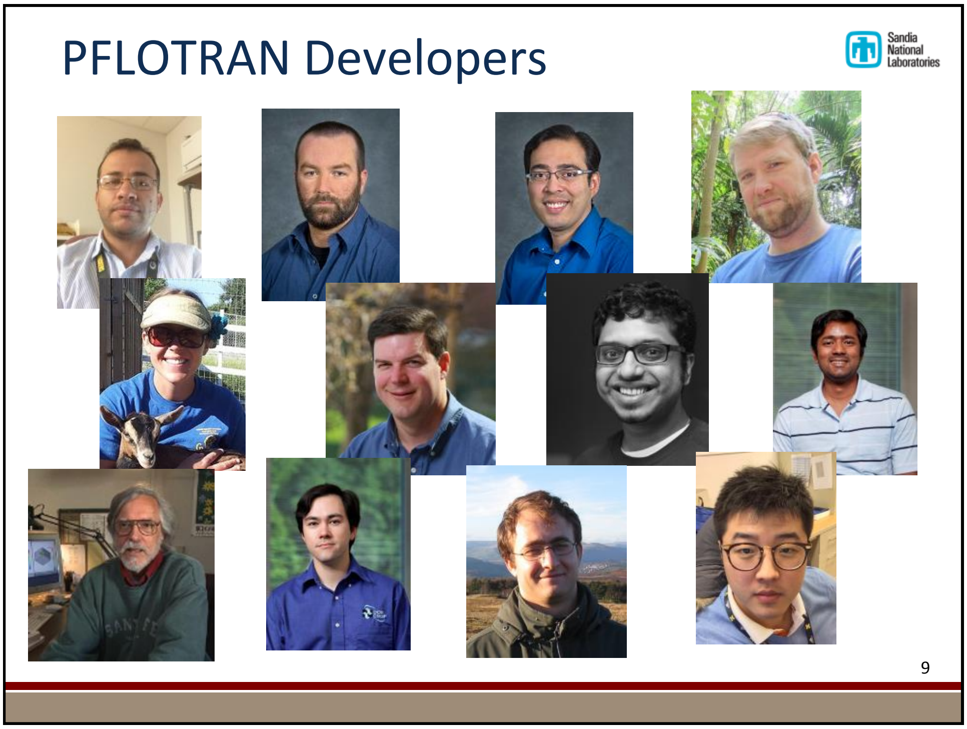
Jitu Kumar

Heeho Park

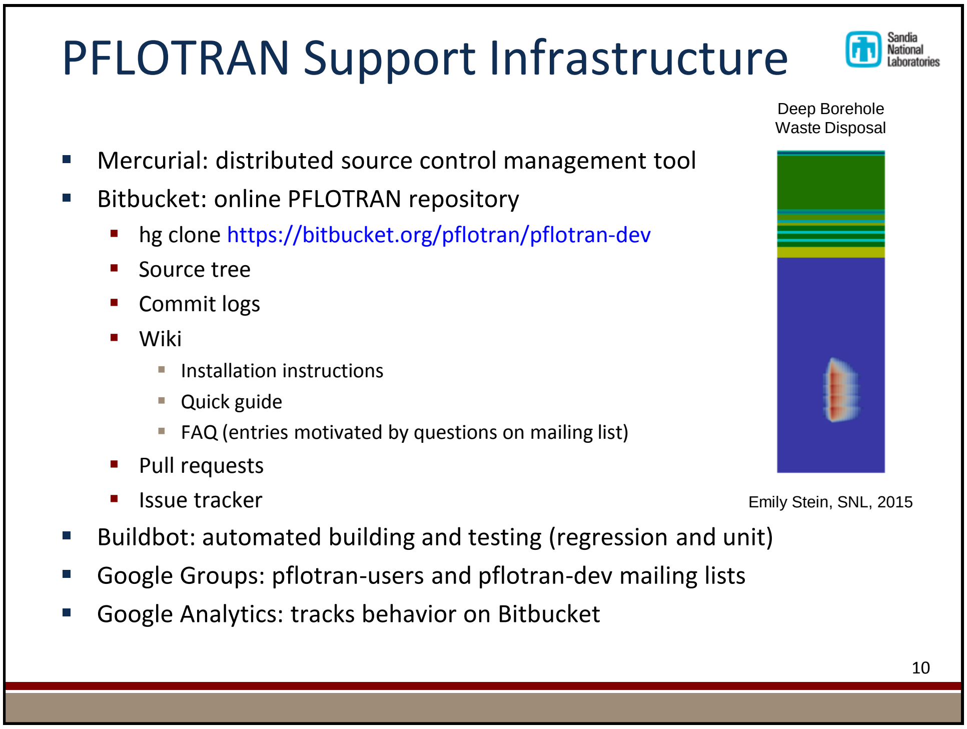
Paolo Orsini

Ayman Alzraiee

Jennifer Frederick



PFLOTRAN Developers



PFLOTRAN Support Infrastructure

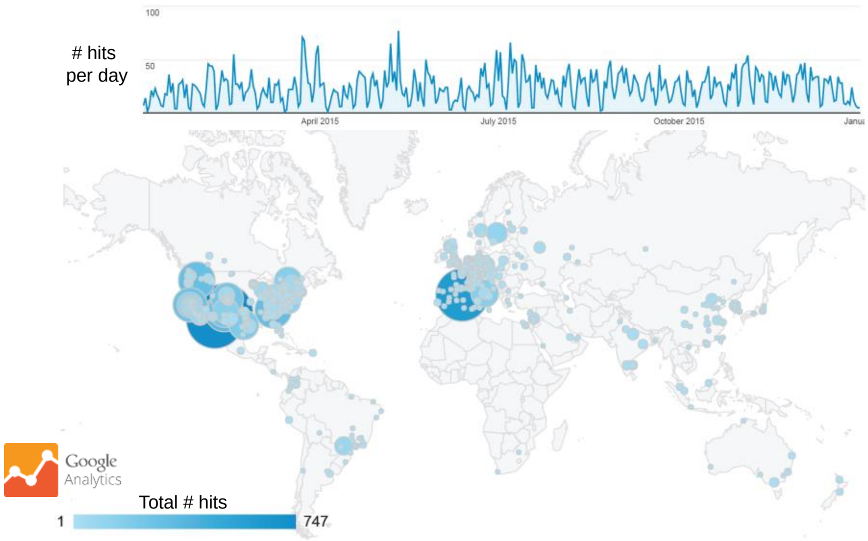
- Mercurial: distributed source control management tool
- Bitbucket: online PFLOTRAN repository
  - hg clone https://bitbucket.org/pflotran/pflotran-dev
  - Source tree
  - Commit logs
  - Wiki
    - Installation instructions
    - Quick guide
    - FAQ (entries motivated by questions on mailing list)
  - Pull requests
  - Issue tracker
- Buildbot: automated building and testing (regression and unit)
- Google Groups: pflotran-users and pflotran-dev mailing lists
- Google Analytics: tracks behavior on Bitbucket

Deep Borehole Waste Disposal

Emily Stein, SNL, 2015

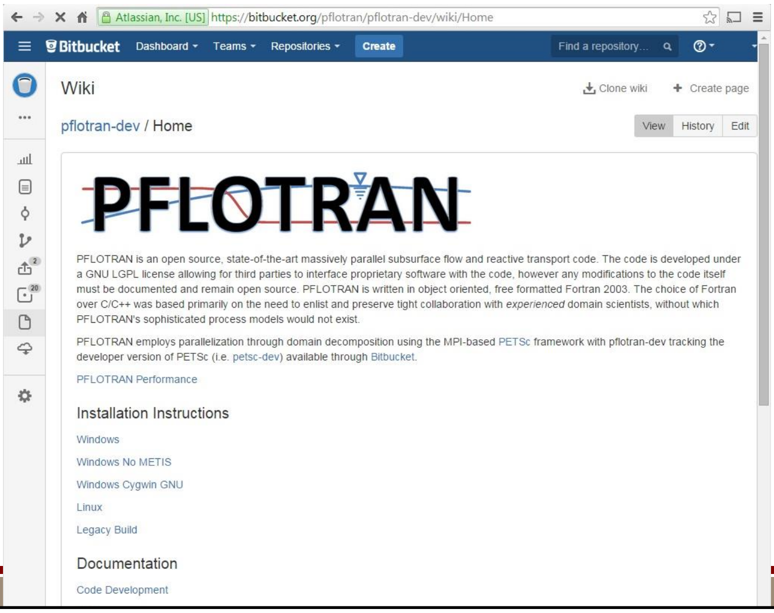
Hits on PFLOTRAN Bitbucket Site in 2015

Sandia National Laboratories



PFLOTRAN Bitbucket Site

Sandia National Laboratories



PFLOTRAN is an open source, state-of-the-art massively parallel subsurface flow and reactive transport code. The code is developed under a GNU LGPL license allowing for third parties to interface proprietary software with the code, however any modifications to the code itself must be documented and remain open source. PFLOTRAN is written in object oriented, free formatted Fortran 2003. The choice of Fortran over C/C++ was based primarily on the need to enlist and preserve tight collaboration with *experienced* domain scientists, without which PFLOTRAN's sophisticated process models would not exist.

PFLOTRAN employs parallelization through domain decomposition using the MPI-based PETSc framework with pflotran-dev tracking the developer version of PETSc (i.e. petsc-dev) available through Bitbucket.

PFLOTRAN Performance

Installation Instructions

Windows

Windows No METIS

Windows Cygwin GNU

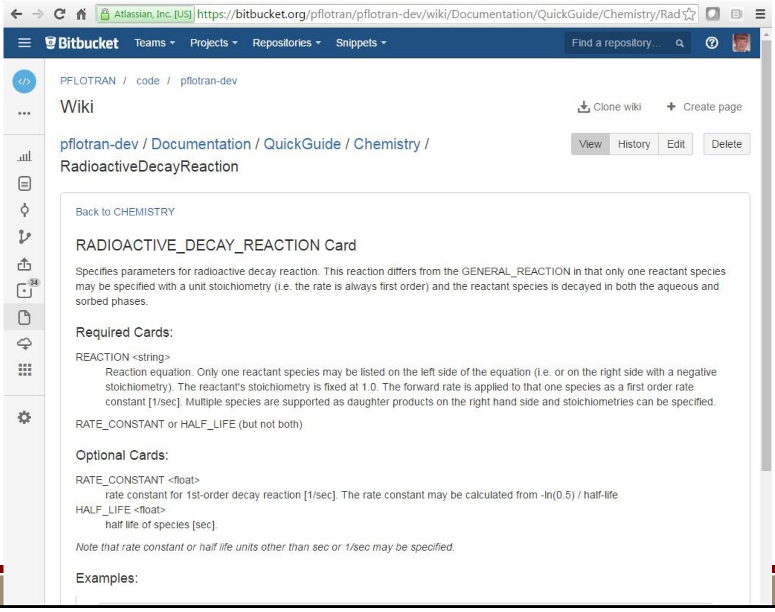
Linux

Legacy Build

Documentation

Code Development

PFLOTRAN Quick Guide


Atlassian, Inc. [US] https://bitbucket.org/pflotran/pflotran-dev/wiki/Documentation/QuickGuide/Chemistry/Rad
Bitbucket
Teams
Projects
Repositories
Snippets
Find a repository...
34
PFLOTRAN / code / pflotran-dev
Wiki
Clone wiki
Create page
View
History
Edit
Delete

pflotran-dev / Documentation / QuickGuide / Chemistry / RadioactiveDecayReaction

Back to CHEMISTRY

RADIOACTIVE_DECAY_REACTION Card

Specifies parameters for radioactive decay reaction. This reaction differs from the GENERAL_REACTION in that only one reactant species may be specified with a unit stoichiometry (i.e. the rate is always first order) and the reactant species is decayed in both the aqueous and sorbed phases.

Required Cards:

REACTION <string>
    Reaction equation. Only one reactant species may be listed on the left side of the equation (i.e. or on the right side with a negative stoichiometry). The reactant's stoichiometry is fixed at 1.0. The forward rate is applied to that one species as a first order rate constant [1/sec]. Multiple species are supported as daughter products on the right hand side and stoichiometries can be specified.

RATE_CONSTANT or HALF_LIFE (but not both)

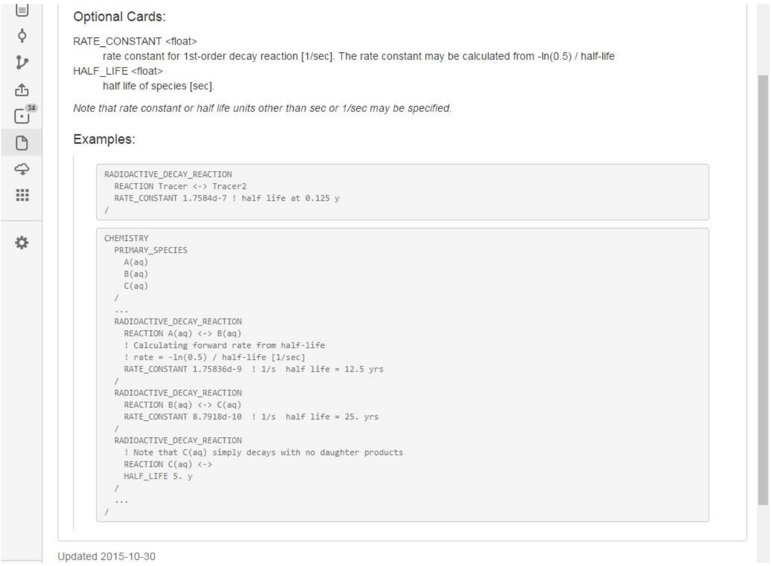
Optional Cards:

RATE_CONSTANT <float>
    rate constant for 1st-order decay reaction [1/sec]. The rate constant may be calculated from -ln(0.5) / half-life
HALF_LIFE <float>
    half life of species [sec].

Note that rate constant or half life units other than sec or 1/sec may be specified.

Examples:

PFLOTRAN Quick Guide (cont.)



Optional Cards:

RATE_CONSTANT <float>
    rate constant for 1st-order decay reaction [1/sec]. The rate constant may be calculated from -ln(0.5) / half-life
HALF_LIFE <float>
    half life of species [sec].

Note that rate constant or half life units other than sec or 1/sec may be specified.

Examples:

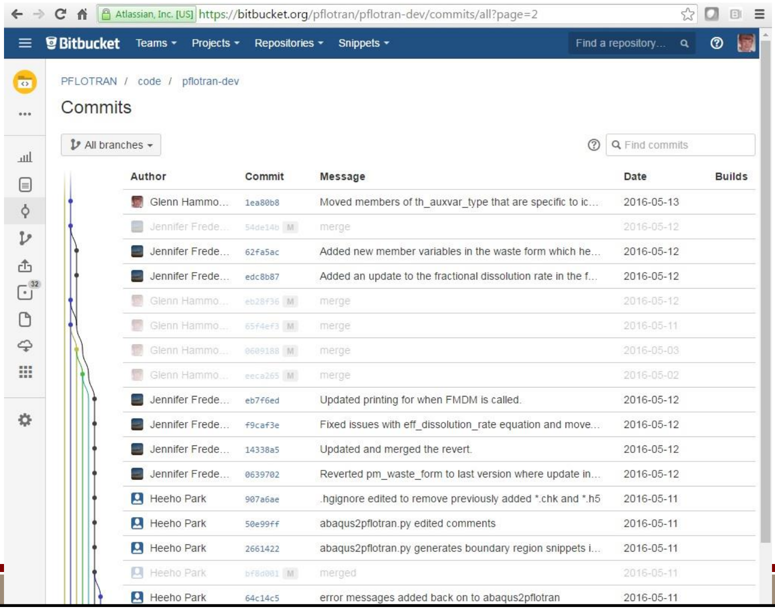
```
RADIOACTIVE_DECAY_REACTION
  REACTION Tracer <-> Tracer2
  RATE_CONSTANT 1.7584d-7 ! half life at 0.125 y
/
```

```
CHEMISTRY
  PRIMARY_SPECIES
    A(aq)
    B(aq)
    C(aq)
  /
  ...
  RADIOACTIVE_DECAY_REACTION
    REACTION A(aq) <-> B(aq)
    ! Calculating forward rate from half-life
    ! rate = -ln(0.5) / half-life [1/sec]
    RATE_CONSTANT 1.75836d-9  ! 1/s  half life = 12.5 yrs
  /
  RADIOACTIVE_DECAY_REACTION
    REACTION B(aq) <-> C(aq)
    RATE_CONSTANT 8.7918d-10  ! 1/s  half life = 25. yrs
  /
  RADIOACTIVE_DECAY_REACTION
    ! Note that C(aq) simply decays with no daughter products
    REACTION C(aq) <->
    HALF_LIFE 5. y
  /
  ...
/
```

Updated 2015-10-30

PFLOTRAN Bitbucket Commit Log

Sandia National Laboratories



Atlassian, Inc. [US] https://bitbucket.org/pflotran/pflotran-dev/commits/all?page=2
Bitbucket
Teams
Projects
Repositories
Snippets
Find a repository…
32

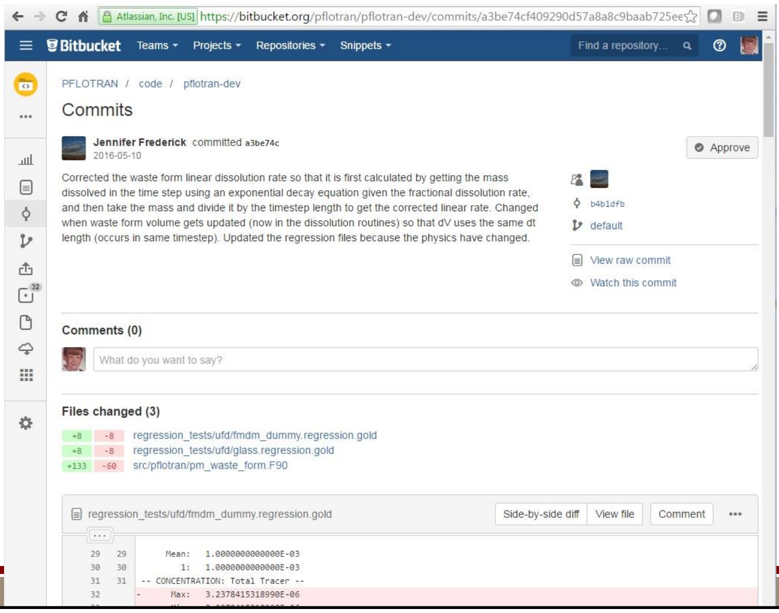
PFLOTRAN / code / pflotran-dev

Commits

Author	Commit	Message	Date	Builds
Glenn Hammo…	1ea80b8	Moved members of th_auxvar_type that are specific to ic…	2016-05-13	
Jennifer Frede…	54de14b M	merge	2016-05-12	
Jennifer Frede…	62fa5ac	Added new member variables in the waste form which he…	2016-05-12	
Jennifer Frede…	edc8b87	Added an update to the fractional dissolution rate in the f…	2016-05-12	
Glenn Hammo…	eb28f36 M	merge	2016-05-12	
Glenn Hammo…	65f4ef3 M	merge	2016-05-11	
Glenn Hammo…	0609188 M	merge	2016-05-03	
Glenn Hammo…	eeca265 M	merge	2016-05-02	
Jennifer Frede…	eb7f6ed	Updated printing for when FMDM is called.	2016-05-12	
Jennifer Frede…	f9caf3e	Fixed issues with eff_dissolution_rate equation and move…	2016-05-12	
Jennifer Frede…	14338a5	Updated and merged the revert.	2016-05-12	
Jennifer Frede…	0639702	Reverted pm_waste_form to last version where update in…	2016-05-12	
Heeho Park	907a6ae	.hgignore edited to remove previously added *.chk and *.h5	2016-05-11	
Heeho Park	50e99ff	abaqus2pflotran.py edited comments	2016-05-11	
Heeho Park	2661422	abaqus2pflotran.py generates boundary region snippets i…	2016-05-11	
Heeho Park	bf8d001 M	merged	2016-05-11	
Heeho Park	64c14c5	error messages added back on to abaqus2pflotran	2016-05-11	

PFLOTRAN Bitbucket Commit

Sandia National Laboratories



Atlassian, Inc. [US] https://bitbucket.org/pflotran/pflotran-dev/commits/a3be74cf409290d57a8a8c9baab725ee
Bitbucket
Teams
Projects
Repositories
Snippets
Find a repository…
32

PFLOTRAN / code / pflotran-dev

Commits

Jennifer Frederick committed a3be74c
2016-05-10

Corrected the waste form linear dissolution rate so that it is first calculated by getting the mass dissolved in the time step using an exponential decay equation given the fractional dissolution rate, and then take the mass and divide it by the timestep length to get the corrected linear rate. Changed when waste form volume gets updated (now in the dissolution routines) so that dV uses the same dt length (occurs in same timestep). Updated the regression files because the physics have changed.

b4b1dfb
default

Comments (0)

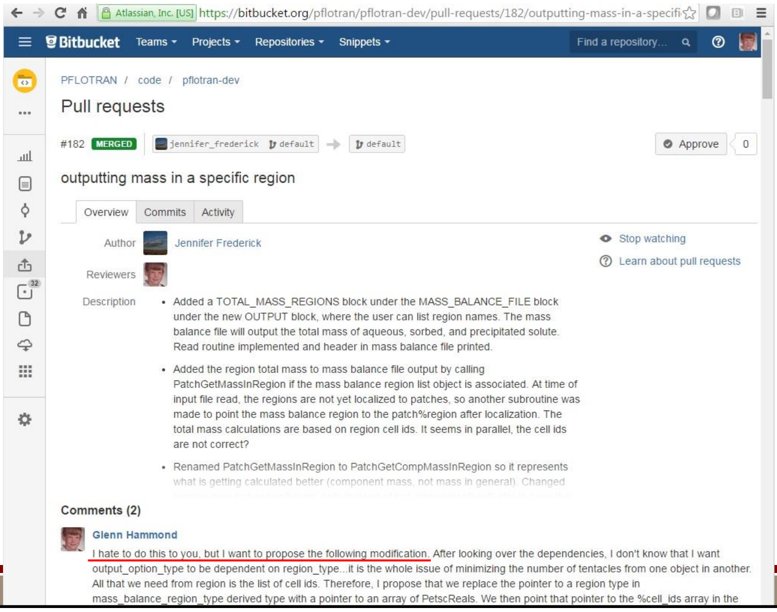
Files changed (3)

+8	-8	regression_tests/ufd/fmdm_dummy.regression.gold
+8	-8	regression_tests/ufd/glass.regression.gold
+133	-60	src/pflotran/pm_waste_form.F90

regression_tests/ufd/fmdm_dummy.regression.gold

```
      Mean:   1.0000000000000E-03
         1:   1.0000000000000E-03
 -- CONCENTRATION: Total Tracer --
-      Max:   3.2378415318990E-06
```

PFLOTRAN Bitbucket Pull Request



Atlassian, Inc. [US] https://bitbucket.org/pflotran/pflotran-dev/pull-requests/182/outputting-mass-in-a-specifi
Bitbucket
Teams
Projects
Repositories
Snippets
Find a repository...
PFLOTRAN / code / pflotran-dev

Pull requests

#182 MERGED jennifer_frederick default → default

outputting mass in a specific region

Overview
Commits
Activity
Stop watching
Learn about pull requests

Author: Jennifer Frederick

Reviewers

Description

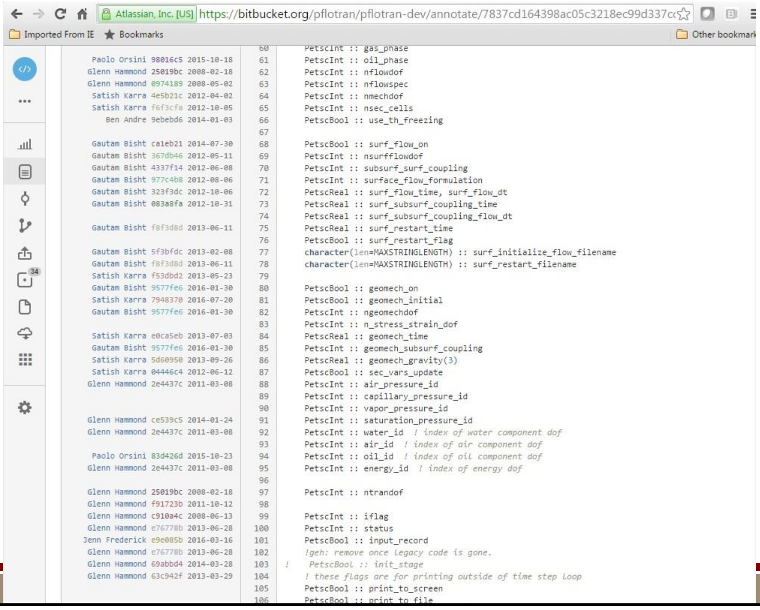
- Added a TOTAL_MASS_REGIONS block under the MASS_BALANCE_FILE block under the new OUTPUT block, where the user can list region names. The mass balance file will output the total mass of aqueous, sorbed, and precipitated solute. Read routine implemented and header in mass balance file printed.
- Added the region total mass to mass balance file output by calling PatchGetMassInRegion if the mass balance region list object is associated. At time of input file read, the regions are not yet localized to patches, so another subroutine was made to point the mass balance region to the patch%region after localization. The total mass calculations are based on region cell ids. It seems in parallel, the cell ids are not correct?
- Renamed PatchGetMassInRegion to PatchGetCompMassInRegion so it represents what is getting calculated better (component mass, not mass in general). Changed

Comments (2)

Glenn Hammond

I hate to do this to you, but I want to propose the following modification. After looking over the dependencies, I don't know that I want output_option_type to be dependent on region_type...it is the whole issue of minimizing the number of tentacles from one object in another. All that we need from region is the list of cell ids. Therefore, I propose that we replace the pointer to a region type in mass_balance_region_type derived type with a pointer to an array of PetscReals. We then point that pointer to the %cell_ids array in the

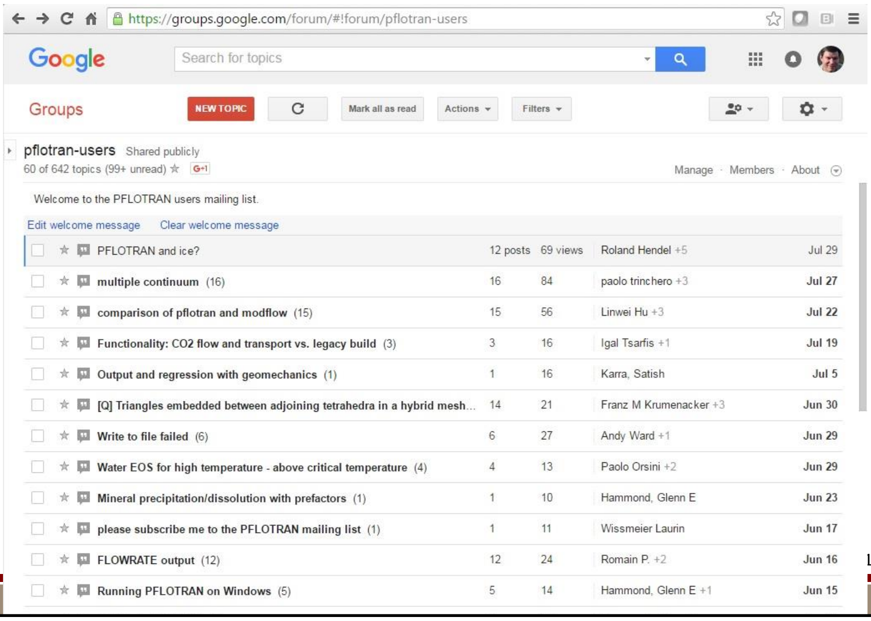
PFLOTRAN Bitbucket Blame



Atlassian, Inc. [US] https://bitbucket.org/pflotran/pflotran-dev/annotate/7837cd164398ac05c3218ec99d337cc
Imported From IE
Bookmarks
Other bookmarks

```
                                          PetscInt :: gas_phase
   Paolo Orsini 98016c5 2015-10-18        PetscInt :: oil_phase
  Glenn Hammond 25019bc 2008-02-18        PetscInt :: nflowdof
  Glenn Hammond 0974189 2008-05-02        PetscInt :: nflowspec
   Satish Karra 4e5b21c 2012-04-02        PetscInt :: nmechdof
   Satish Karra f6f3cfa 2012-10-05        PetscInt :: nsec_cells
     Ben Andre 9ebebd6 2014-01-03         PetscBool :: use_th_freezing

   Gautam Bisht ca1eb21 2014-07-30        PetscBool :: surf_flow_on
   Gautam Bisht 367db46 2012-05-11        PetscInt :: nsurfflowdof
   Gautam Bisht 4337f14 2012-06-08        PetscInt :: subsurf_surf_coupling
   Gautam Bisht 977c4b8 2012-08-06        PetscInt :: surface_flow_formulation
   Gautam Bisht 323f3dc 2012-10-06        PetscReal :: surf_flow_time, surf_flow_dt
   Gautam Bisht 083a8fa 2012-10-31        PetscReal :: surf_subsurf_coupling_time
                                          PetscReal :: surf_subsurf_coupling_flow_dt
   Gautam Bisht f8f3d8d 2013-06-11        PetscReal :: surf_restart_time
                                          PetscBool :: surf_restart_flag
   Gautam Bisht 5f3bfdc 2013-02-08        character(len=MAXSTRINGLENGTH) :: surf_initialize_flow_filename
   Gautam Bisht f8f3d8d 2013-06-11        character(len=MAXSTRINGLENGTH) :: surf_restart_filename
   Satish Karra f53dbd2 2013-05-23
   Gautam Bisht 9577fe6 2016-01-30        PetscBool :: geomech_on
   Satish Karra 7948370 2016-07-20        PetscBool :: geomech_initial
   Gautam Bisht 9577fe6 2016-01-30        PetscInt :: ngeomechdof
                                          PetscInt :: n_stress_strain_dof
   Satish Karra e0ca5eb 2013-07-03        PetscReal :: geomech_time
   Gautam Bisht 9577fe6 2016-01-30        PetscInt :: geomech_subsurf_coupling
   Satish Karra 5d60950 2013-09-26        PetscReal :: geomech_gravity(3)
   Satish Karra 04446c4 2012-06-12        PetscBool :: sec_vars_update
  Glenn Hammond 2e4437c 2011-03-08        PetscInt :: air_pressure_id
                                          PetscInt :: capillary_pressure_id
                                          PetscInt :: vapor_pressure_id
  Glenn Hammond ce539c5 2014-01-24        PetscInt :: saturation_pressure_id
  Glenn Hammond 2e4437c 2011-03-08        PetscInt :: water_id  ! index of water component dof
                                          PetscInt :: air_id  ! index of air component dof
   Paolo Orsini 83d426d 2015-10-23        PetscInt :: oil_id  ! index of oil component dof
  Glenn Hammond 2e4437c 2011-03-08        PetscInt :: energy_id  ! index of energy dof

  Glenn Hammond 25019bc 2008-02-18        PetscInt :: ntrandof
  Glenn Hammond f91723b 2011-10-12
  Glenn Hammond c910a4c 2008-06-13        PetscInt :: iflag
  Glenn Hammond e76778b 2013-06-28        PetscInt :: status
 Jenn Frederick e9e085b 2016-03-16        PetscBool :: input_record
  Glenn Hammond e76778b 2013-06-28        !geh: remove once legacy code is gone.
  Glenn Hammond 69abbd4 2014-03-28      !    PetscBool :: init_stage
  Glenn Hammond 63c942f 2013-03-29        ! these flags are for printing outside of time step loop
                                          PetscBool :: print_to_screen
                                          PetscBool :: print_to_file
```

PFLOTRAN User Support



pflotran-users Shared publicly

60 of 642 topics (99+ unread)

Welcome to the PFLOTRAN users mailing list.

Edit welcome message Clear welcome message

Topic	Posts	Views	Authors	Date
PFLOTRAN and ice?	12 posts	69 views	Roland Hendel +5	Jul 29
multiple continuum (16)	16	84	paolo trinchero +3	Jul 27
comparison of pflotran and modflow (15)	15	56	Linwei Hu +3	Jul 22
Functionality: CO2 flow and transport vs. legacy build (3)	3	16	Igal Tsarfis +1	Jul 19
Output and regression with geomechanics (1)	1	16	Karra, Satish	Jul 5
[Q] Triangles embedded between adjoining tetrahedra in a hybrid mesh...	14	21	Franz M Krumenacker +3	Jun 30
Write to file failed (6)	6	27	Andy Ward +1	Jun 29
Water EOS for high temperature - above critical temperature (4)	4	13	Paolo Orsini +2	Jun 29
Mineral precipitation/dissolution with prefactors (1)	1	10	Hammond, Glenn E	Jun 23
please subscribe me to the PFLOTRAN mailing list (1)	1	11	Wissmeier Laurin	Jun 17
FLOWRATE output (12)	12	24	Romain P. +2	Jun 16
Running PFLOTRAN on Windows (5)	5	14	Hammond, Glenn E +1	Jun 15

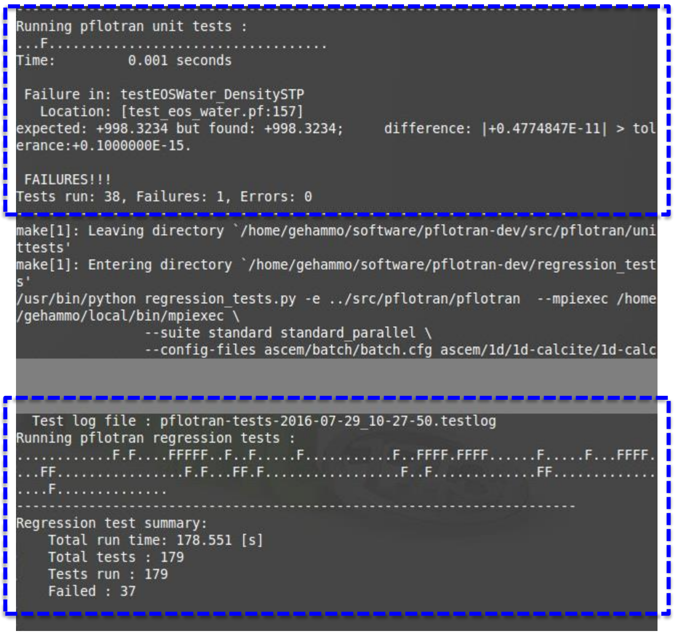
Testing

- Verification: Results match analytical solutions or other results from other simulators
- Validation: Results are correct
- Unit: Chunks of code return the expected result
  - Constitutive relations: capillary pressure, saturation, relative permeability functions
  - Equations of state: density, enthalpy, viscosity, etc.
- Regression: Entire simulation returns the expected result
  - Flow, transport, reaction
  - Structured, unstructured grids

Testing (cont.)

- Example regression test failure
  - Perturb critical pressure for water equation of state by 10 billionths of a percent

```
diff -r f9f01bbf557a src/pflotran/eos_water.F90
--- a/src/pflotran/eos_water.F90     Thu Jul 28 18:59:00 2016 -0700
+++ b/src/pflotran/eos_water.F90     Fri Jul 29 10:31:57 2016 -0700
@@ -893,6 +893,7 @@

   tc1 = H2O_CRITICAL_TEMPERATURE    ! K
   pc1 = H2O_CRITICAL_PRESSURE       ! Pa
+  pc1 = pc1 + 1.d-10*H2O_CRITICAL_PRESSURE ! perturb by 1e-10
   vc1 = 0.00317d0  ! m^3/kg
   utc1 = one/tc1   ! 1/C
   upc1 = one/pc1   ! 1/Pa
```



Unit

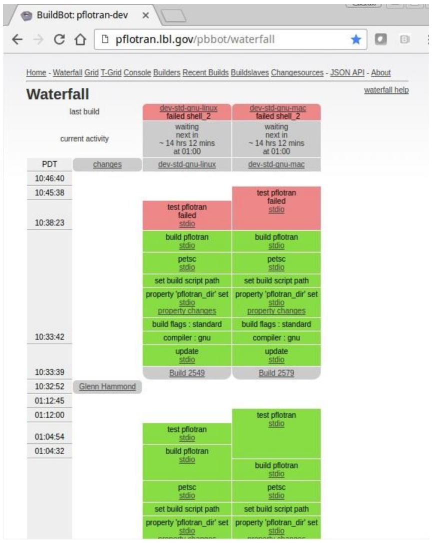
```
Running pflotran unit tests :
...F.................................
Time:         0.001 seconds

 Failure in: testEOSWater_DensitySTP
   Location: [test_eos_water.pf:157]
expected: +998.3234 but found: +998.3234;     difference: |+0.4774847E-11| > tol
erance:+0.1000000E-15.

 FAILURES!!!
Tests run: 38, Failures: 1, Errors: 0
make[1]: Leaving directory `/home/gehammo/software/pflotran-dev/src/pflotran/uni
ttests'
make[1]: Entering directory `/home/gehammo/software/pflotran-dev/regression_test
s'
/usr/bin/python regression_tests.py -e ../src/pflotran/pflotran  --mpiexec /home
/gehammo/local/bin/mpiexec \
                --suite standard standard_parallel \
                --config-files ascem/batch/batch.cfg ascem/1d/1d-calcite/1d-calc
```

Regression

```
  Test log file : pflotran-tests-2016-07-29_10-27-50.testlog
Running pflotran regression tests :
............F.F....FFFFF..F..F.....F...........F..FFFF.FFFF......F.....F...FFFF.
...FF................F.F...FF.F.................F..F.............FF.............
....F..............
----------------------------------------------------------------------
Regression test summary:
    Total run time: 178.551 [s]
    Total tests : 179
    Tests run : 179
    Failed : 37
```

Buildbot: pflotran.lbl.gov/pbbot/waterfall

Sandia National Laboratories



Home - Waterfall Grid T-Grid Console Builders Recent Builds Buildslaves Changesources - JSON API - About

Waterfall

waterfall help

last build		dev-std-gnu-linux failed shell_2	dev-std-gnu-mac failed shell_2
current activity		waiting next in ~ 14 hrs 12 mins at 01:00	waiting next in ~ 14 hrs 12 mins at 01:00
PDT	changes	dev-std-gnu-linux	dev-std-gnu-mac
10:46:40			
10:45:38			test pflotran failed stdio
10:38:23		test pflotran failed stdio	
		build pflotran stdio	build pflotran stdio
		petsc stdio	petsc stdio
		set build script path	set build script path
		property 'pflotran_dir' set stdio property changes	property 'pflotran_dir' set stdio property changes
		build flags : standard	build flags : standard
10:33:42		compiler : gnu	compiler : gnu
		update stdio	update stdio
10:33:39		Build 2549	Build 2579
10:32:52	Glenn Hammond		
01:12:45			
01:12:00			test pflotran stdio
01:04:54		test pflotran stdio	
01:04:32		build pflotran stdio	
			build pflotran stdio
		petsc stdio	petsc stdio
		set build script path	set build script path
		property 'pflotran_dir' set stdio	property 'pflotran_dir' set stdio