

*Exceptional service in the national interest*



www.documentation.pflotran.org

PFLOTRAN Documentation

**PFLOTRAN**

PFLOTRAN is on Bitbucket!

Click here to download a pdf version of the PFLOTRAN documentation.

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

Travis CI

pflotran / pflotran

Build Jobs

Build Job	Status	Duration
master - changed in testing status, will create build...	passed	Run for 24 min 33 sec
develop - build failed	failed	Total time: 10 min 23 sec
develop - build failed	failed	Run for 12 min 12 sec
develop - build failed	failed	Run for 12 min 12 sec
develop - build failed	failed	Run for 12 min 12 sec

Bitbucket

pflotran / pflotran

Wiki

pflotran / Home

**PFLOTRAN**

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 git master branch of PETSc available through Bitbucket. Please visit our website, pflotran.org, for more information. **Looking for the old wiki?** The old wiki can

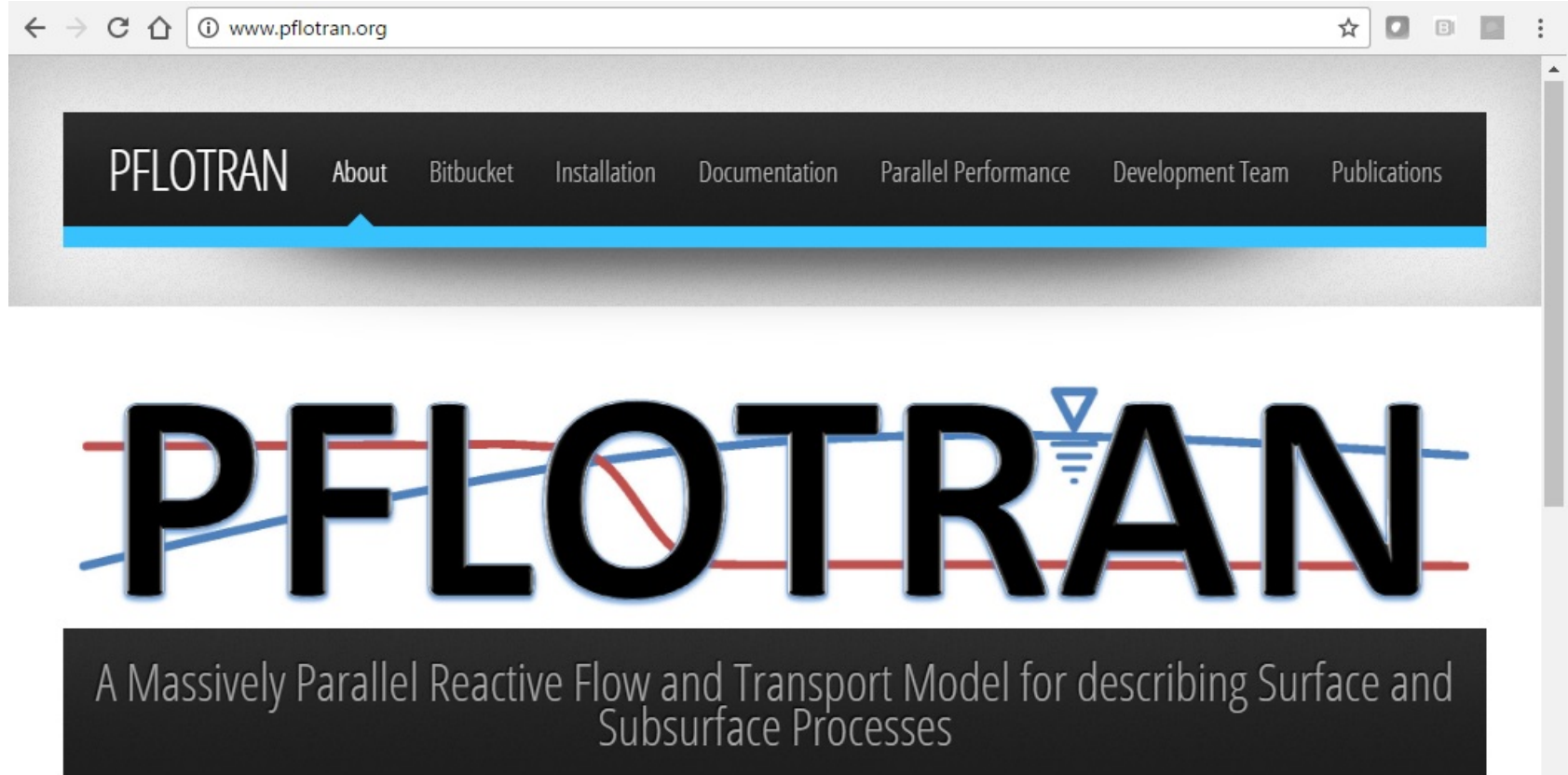
# PFLOTRAN Support Infrastructure



Sandia National Laboratories is a multi-program 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. SAND NO. 2011-XXXXP

# PFLOTRAN Support Infrastructure

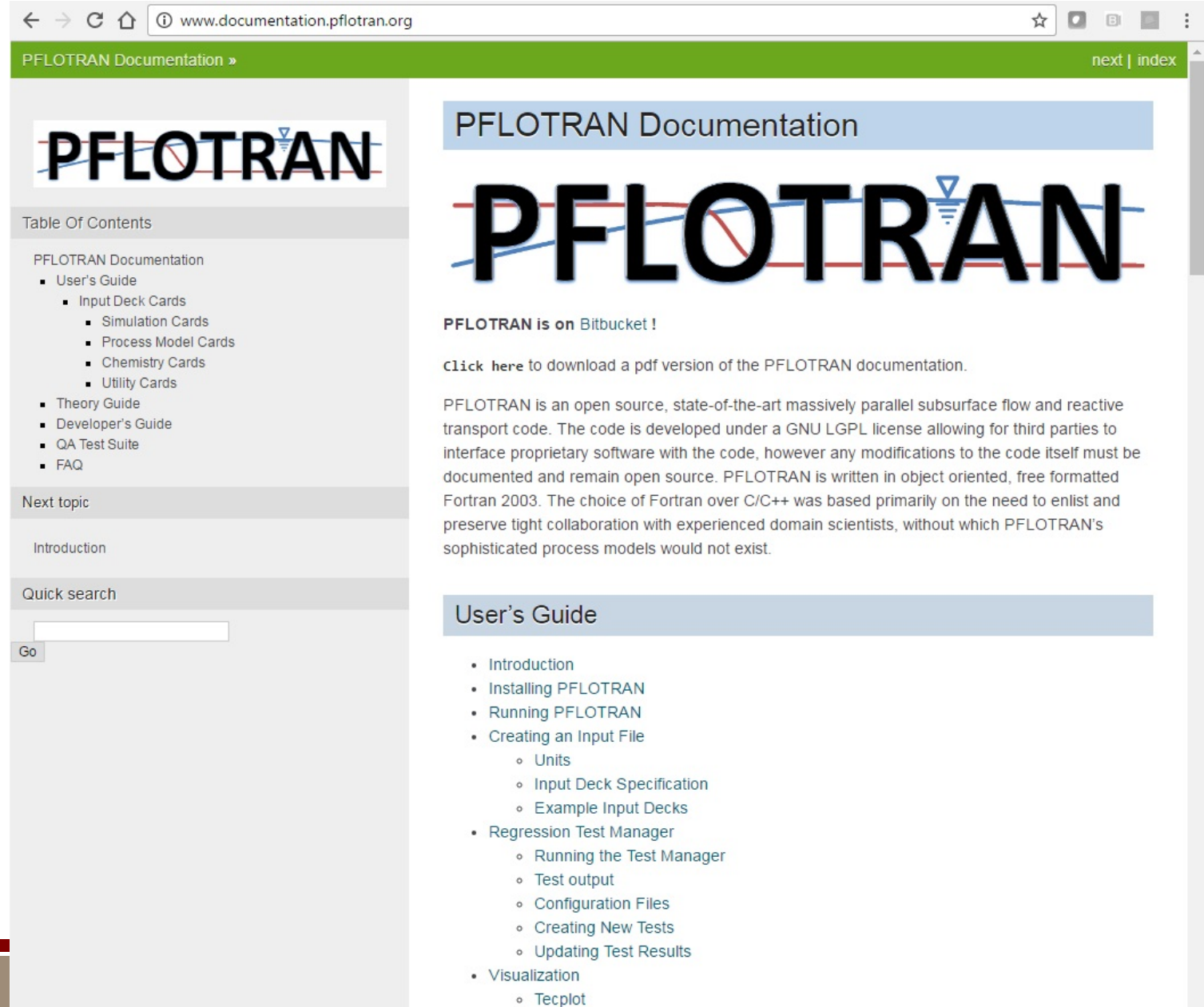
- [www.pflotran.org](http://www.pflotran.org): documentation and overview
- Git: distributed source control management tool
- Bitbucket: online PFLOTRAN repository
  - git clone <https://bitbucket.org/pflotran/pflotran>
  - Source tree
  - Commit logs
  - Pull requests
  - Issue tracker
  - Wiki
    - Link to [documentation.pflotran.org](http://documentation.pflotran.org)
      - Developer/Theory/User guides
      - FAQ (entries motivated by questions on mailing list)
- Travis CI: automated building and testing (regression and unit)
- Google Groups: [pflotran-users](#) and [pflotran-dev](#) mailing lists
- Google Analytics: tracks behavior on Bitbucket



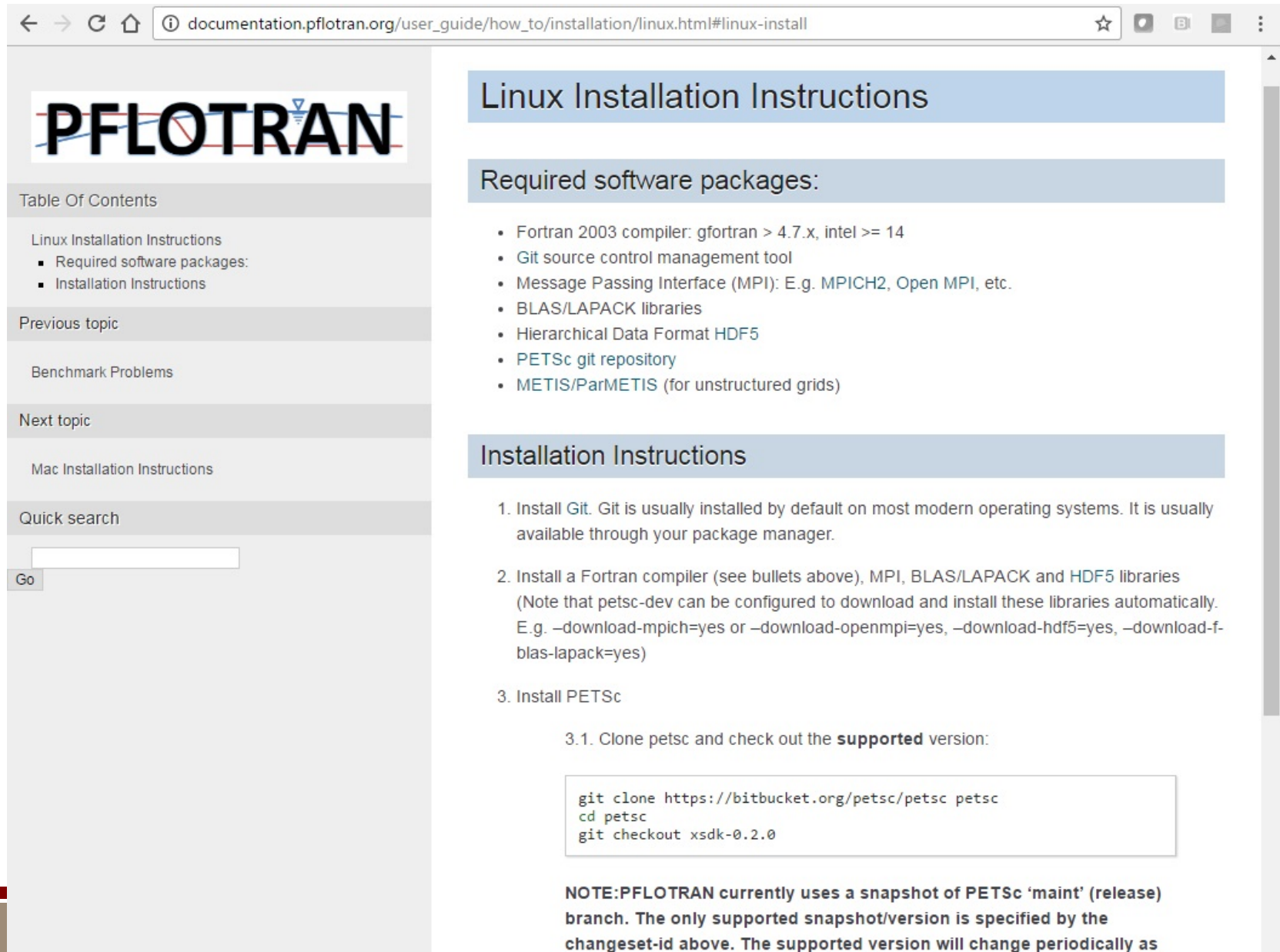
The screenshot shows a web browser window with the address bar displaying "www.pflotran.org". The website has a dark navigation bar with the following links: "PFLOTRAN", "About", "Bitbucket", "Installation", "Documentation", "Parallel Performance", "Development Team", and "Publications". Below the navigation bar is a large graphic of the word "PFLOTRAN" in a bold, black, sans-serif font. The letters are overlaid with a blue line graph showing a fluctuating trend. A red line is also visible, crossing the blue line. Below the graphic is a dark grey box containing the text: "A Massively Parallel Reactive Flow and Transport Model for describing Surface and Subsurface Processes".

## What is PFLOTRAN?

PFLOTRAN is an open source, state-of-the-art massively parallel subsurface flow and reactive transport code. PFLOTRAN solves a system of generally nonlinear partial differential equations describing multiphase, multicomponent and multiscale reactive flow and transport in porous materials. The code is designed to run on massively parallel computing architectures as well as workstations and laptops. Parallelization is achieved through domain decomposition using the [PETSc](#) (Portable Extensible Toolkit for Scientific Computation) libraries. PFLOTRAN has been developed from the ground up for parallel scalability and has been run on up to  $2^{18}$  processor cores with



The screenshot shows a web browser displaying the PFLOTRAN documentation website. The browser's address bar shows the URL [www.documentation.pflotran.org](http://www.documentation.pflotran.org). The website has a green header with the text "PFLOTRAN Documentation" and navigation links for "next" and "index". The main content area features a large "PFLOTRAN" logo with a blue and red graphic element. Below the logo, there is a section titled "PFLOTRAN is on Bitbucket!" and a link to download a PDF version of the documentation. A paragraph of text describes PFLOTRAN as an open source, state-of-the-art massively parallel subsurface flow and reactive transport code. The left sidebar contains a "Table Of Contents" with a list of links: "User's Guide", "Input Deck Cards" (with sub-links for "Simulation Cards", "Process Model Cards", "Chemistry Cards", and "Utility Cards"), "Theory Guide", "Developer's Guide", "QA Test Suite", and "FAQ". Below the table of contents is a "Next topic" section with a link to "Introduction". At the bottom of the sidebar is a "Quick search" section with a search input field and a "Go" button. The main content area also has a "User's Guide" section with a list of links: "Introduction", "Installing PFLOTRAN", "Running PFLOTRAN", "Creating an Input File" (with sub-links for "Units", "Input Deck Specification", and "Example Input Decks"), "Regression Test Manager" (with sub-links for "Running the Test Manager", "Test output", "Configuration Files", "Creating New Tests", and "Updating Test Results"), and "Visualization" (with a sub-link for "Tecplot").



The screenshot shows a web browser window with the URL `documentation.pflotran.org/user_guide/how_to/installation/linux.html#linux-install`. The page features a sidebar on the left with the Pflotran logo and navigation links for 'Table Of Contents', 'Previous topic', 'Next topic', and a 'Quick search' box. The main content area is titled 'Linux Installation Instructions' and is divided into two sections: 'Required software packages:' and 'Installation Instructions'. The 'Required software packages:' section lists: Fortran 2003 compiler (gfortran > 4.7.x, intel >= 14), Git, Message Passing Interface (MPI), BLAS/LAPACK libraries, Hierarchical Data Format HDF5, PETSc git repository, and METIS/ParMETIS. The 'Installation Instructions' section contains three numbered steps: 1. Install Git, 2. Install a Fortran compiler, MPI, BLAS/LAPACK and HDF5 libraries (with a note about PETSc configuration), and 3. Install PETSc. Step 3.1 includes a code block for cloning and checking out the PETSc repository.

documentation.pflotran.org/user\_guide/how\_to/installation/linux.html#linux-install

## Linux Installation Instructions

### Required software packages:

- Fortran 2003 compiler: gfortran > 4.7.x, intel >= 14
- Git source control management tool
- Message Passing Interface (MPI): E.g. MPICH2, Open MPI, etc.
- BLAS/LAPACK libraries
- Hierarchical Data Format HDF5
- PETSc git repository
- METIS/ParMETIS (for unstructured grids)

### Installation Instructions

1. Install Git. Git is usually installed by default on most modern operating systems. It is usually available through your package manager.
2. Install a Fortran compiler (see bullets above), MPI, BLAS/LAPACK and HDF5 libraries (Note that petsc-dev can be configured to download and install these libraries automatically. E.g. `--download-mpich=yes` or `--download-openmpi=yes`, `--download-hdf5=yes`, `--download-fblas-lapack=yes`)
3. Install PETSc
  - 3.1. Clone petsc and check out the **supported** version:

```
git clone https://bitbucket.org/petsc/petsc petsc
cd petsc
git checkout xsdk-0.2.0
```

**NOTE:**PFLOTRAN currently uses a snapshot of PETSc 'maint' (release) branch. The only supported snapshot/version is specified by the changeset-id above. The supported version will change periodically as



Table Of Contents

- Mode: **REACTIVE TRANSPORT** (Keyword **CHEMISTRY**)
  - Mineral Precipitation and Dissolution
    - Rate Limiter
    - Changes in Material Properties
    - Affinity Threshold
    - Surface Armoring
  - Sorption
    - Ion Exchange
    - Surface Complexation
    - Multirate Sorption
  - Sorption Isotherm <Under Revision>
  - Colloid-Facilitated Transport
  - Tracer Mean Age
  - Thermodynamic Database
  - Eh, pe

Previous topic

Mode: **MISCIBLE**

Next topic

Mode: **GEOMECHANICS**

Quick search

Go

## Mode: REACTIVE TRANSPORT (Keyword CHEMISTRY)

The governing mass conservation equations for the geochemical transport mode for a multiphase system written in terms of a set of independent aqueous primary or basis species with the form

$$\frac{\partial}{\partial t} (\varphi \sum_{\alpha} s_{\alpha} \Psi_j^{\alpha}) + \nabla \cdot \sum_{\alpha} \Omega_j^{\alpha} = Q_j - \sum_m \nu_{jm} I_m - \frac{\partial S_j}{\partial t}, \quad (1)$$

and

$$\frac{\partial \varphi_m}{\partial t} = \bar{V}_m I_m, \quad (2)$$

for minerals with molar volume  $\bar{V}_m$ , mineral reaction rate  $I_m$  and mineral volume fraction  $\varphi_m$  referenced to an REV. Sums over  $\alpha$  in Eqn. (1) are over all fluid phases in the system. The quantity  $\Psi_j^{\alpha}$  denotes the total concentration of the  $j$ th primary species  $\mathcal{A}_j^{\text{pri}}$  in the  $\alpha$ th fluid phase defined by

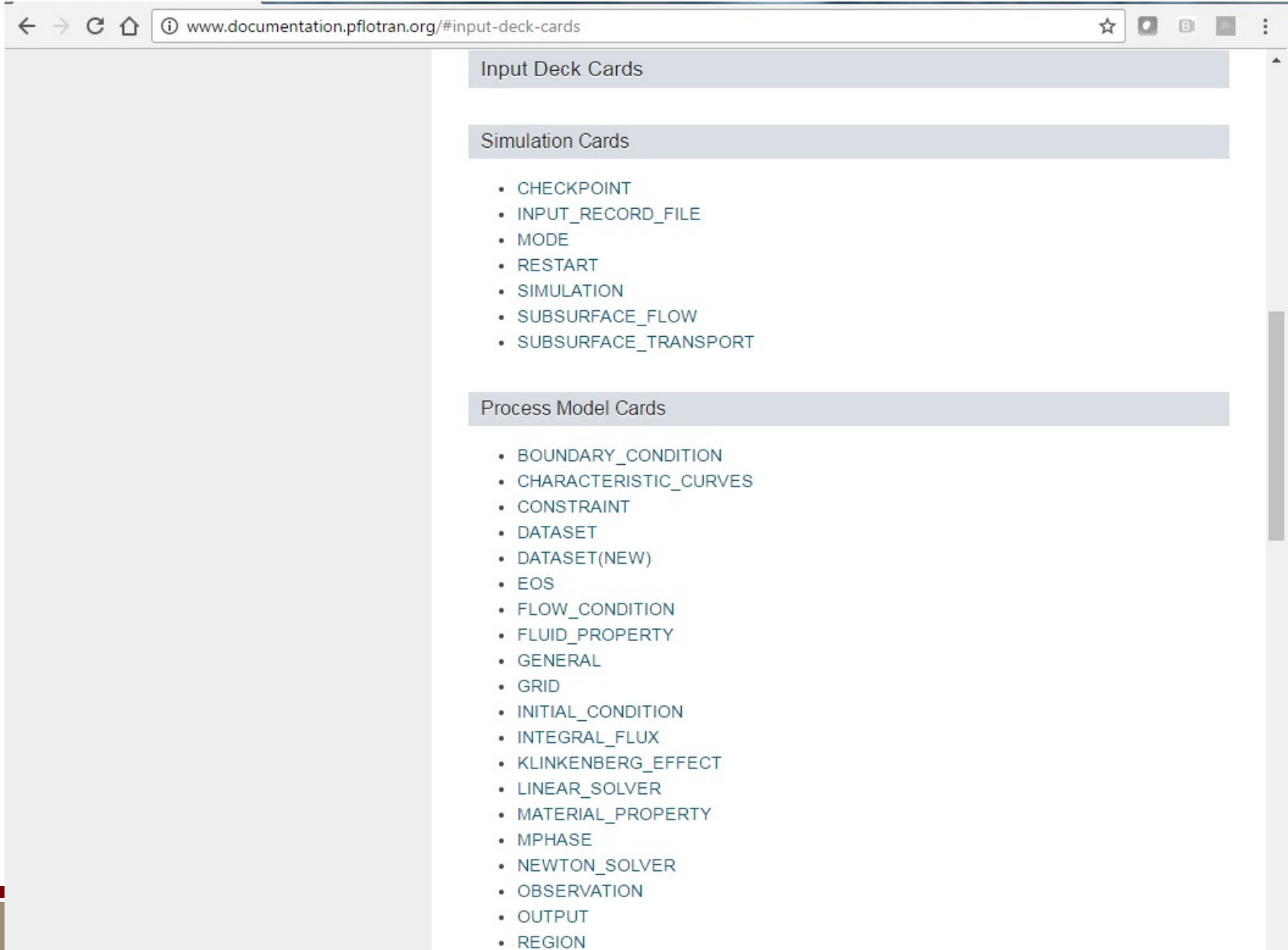
$$\Psi_j^{\alpha} = \delta_{l\alpha} C_j^l + \sum_{i=1}^{N_{\text{sec}}} \nu_{ji}^{\alpha} C_i^{\alpha}, \quad (3)$$

In this equation the subscript  $l$  represents the aqueous electrolyte phase from which the primary species are chosen. The secondary species concentrations  $C_i^{\alpha}$  are obtained from mass action equations corresponding to equilibrium conditions of the reactions

$$\sum_j \nu_{ji}^{\alpha} \mathcal{A}_j^l \rightleftharpoons \mathcal{A}_i^{\alpha}, \quad (4)$$

yielding the mass action equations

$$C_i^{\alpha} = \frac{K_i^{\alpha}}{\gamma_i^{\alpha}} \prod_j (\gamma_j^l C_j^l)^{\nu_{ji}^{\alpha}}, \quad (5)$$



www.documentation.pflotran.org/#input-deck-cards

## Input Deck Cards

### Simulation Cards

- CHECKPOINT
- INPUT\_RECORD\_FILE
- MODE
- RESTART
- SIMULATION
- SUBSURFACE\_FLOW
- SUBSURFACE\_TRANSPORT

### Process Model Cards

- BOUNDARY\_CONDITION
- CHARACTERISTIC\_CURVES
- CONSTRAINT
- DATASET
- DATASET(NEW)
- EOS
- FLOW\_CONDITION
- FLUID\_PROPERTY
- GENERAL
- GRID
- INITIAL\_CONDITION
- INTEGRAL\_FLUX
- KLINKENBERG\_EFFECT
- LINEAR\_SOLVER
- MATERIAL\_PROPERTY
- MPHASE
- NEWTON\_SOLVER
- OBSERVATION
- OUTPUT
- REGION

www.documentation.pflotran.org/user\_guide/cards/process\_model\_cards/time\_card.html

## PFLOTRAN

Table Of Contents

- TIME
  - Required Cards
  - Optional Cards:
  - Examples

Previous topic

TH

Next topic

TIMESTEPPER

Quick search

Go

### TIME

Specifies the time step sizes and final simulation time.

### Required Cards

FINAL\_TIME <float> <string>  
Specified the final time of the simulation with units.

### Optional Cards:

STEADY\_STATE  
Specifies that the simulation run in steady state mode with no time stepping (**Warning: Experimental**)

INITIAL\_TIMESTEP\_SIZE <float> <string>  
Specifies the initial time step size

MINIMUM\_TIMESTEP\_SIZE <float> <string>  
Specifies the minimum time step size. This minimum step size is used to softly exit the simulation when time step sizes are cut below this threshold. If TS\_ACCELERATION from the TIMESTEPPER card is used, the step size will not drop below this threshold.

MAXIMUM\_TIMESTEP\_SIZE <float> <string> <optional string> <optional float> <optional string>  
Specifies the maximum time step size. This maximum time step size can change during the simulation by adding the optional float/strings as follows:

```
MAXIMUM_TIMESTEP_SIZE <float> at <float> <string> (See examples below)
```

### Examples

```
TIME
# total simulation length, the unit can be in h or y
FINAL_TIME 5200.0d0 h
# initial time step size, it could be reduced or increased automatically
# by the program for accuracy purpose
INITIAL_TIMESTEP_SIZE 1.0d0 h
# maximum time step that can not be exceeded
MAXIMUM_TIMESTEP_SIZE 1.0d0 h
END

# to specify an initial maximum time step size of 1 hour, decrease the
# time step size to 0.1 hours between 24 and 34 hours simulation time,
# and revert back to the original time step size afterwards
TIME
FINAL_TIME 96.0d0 h
INITIAL_TIMESTEP_SIZE 0.1d0 h
MAXIMUM_TIMESTEP_SIZE 1.0d0 h at 0. h
MAXIMUM_TIMESTEP_SIZE 0.1d0 h at 24. h
MAXIMUM_TIMESTEP_SIZE 1.0d0 h at 34. h
END
```

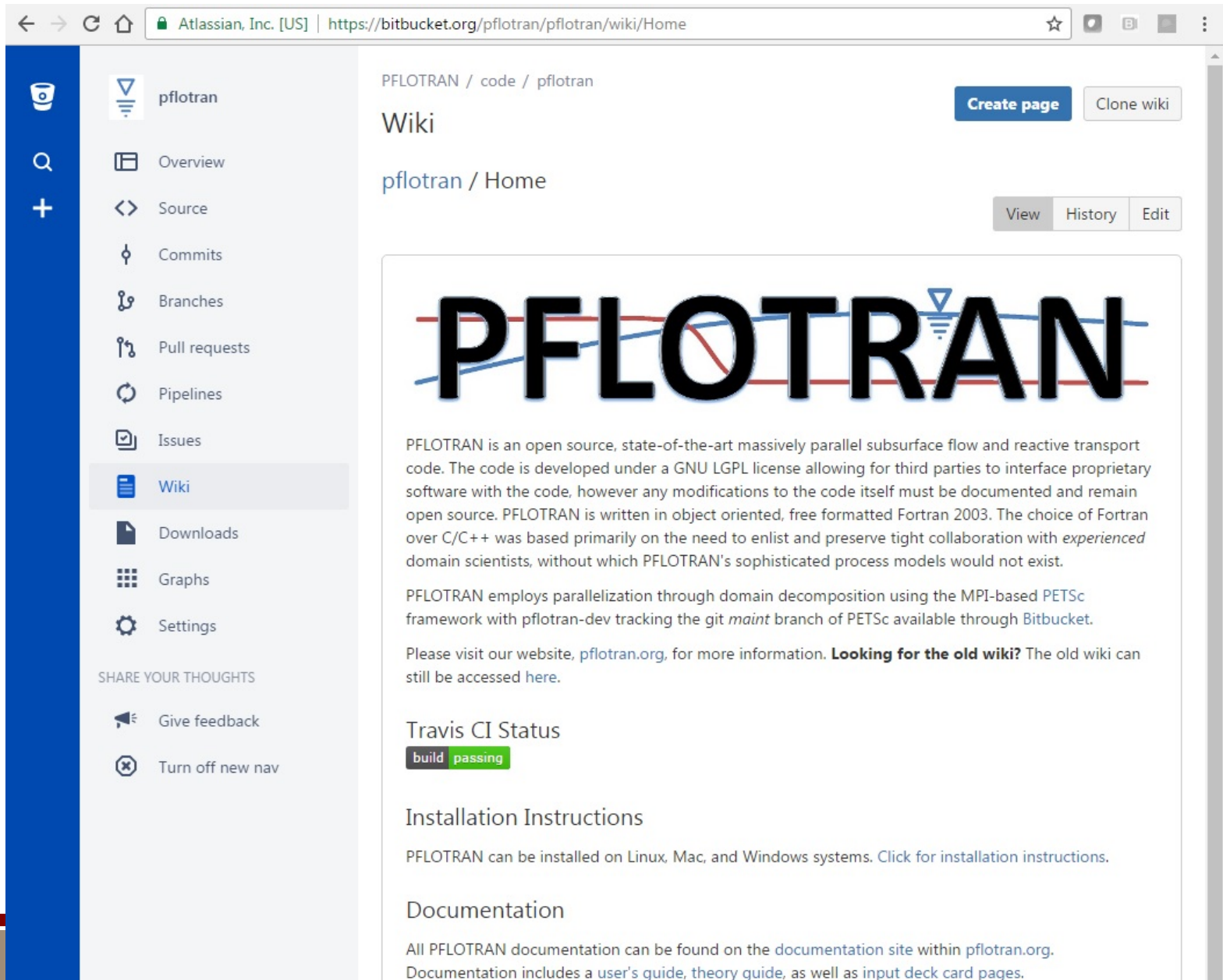
How do I make any sense of the screen output, in particular Newton iteration convergence?

## Standard Flow

```
== GENERAL FLOW =====  
1 2r: 4.13E-13 2x: 1.30E+08 2u: 4.47E-01 ir: 4.42E-14 iu: 2.44E-02 rsn: 0  
2 2r: 6.06E-13 2x: 1.30E+08 2u: 5.63E-01 ir: 5.65E-14 iu: 5.23E-02 rsn: 0  
3 2r: 2.21E-13 2x: 1.30E+08 2u: 3.67E-01 ir: 1.90E-14 iu: 3.05E-02 rsn: 0  
4 2r: 2.45E-13 2x: 1.30E+08 2u: 3.33E-01 ir: 2.20E-14 iu: 2.42E-02 rsn: itol_post_che  
  
Step 5498 Time= 9.89040E+03 Dt= 1.04142E+01 [y] snes_conv_reason: 12  
newton = 4 [ 31891] linear = 75 [ 645652] cuts = 0 [ 915]  
--> SNES Linear/Non-Linear Iterations = 75 / 4  
--> SNES Residual: 2.452466E-13 2.335682E-16 2.200252E-14  
--> max chng: dpl= 2.7064E-03 dpg= 1.1357E-01 dpa= 1.1357E-01  
              dxa= 1.3283E-11 dt= 0.0000E+00 dsg= 2.9327E-08
```

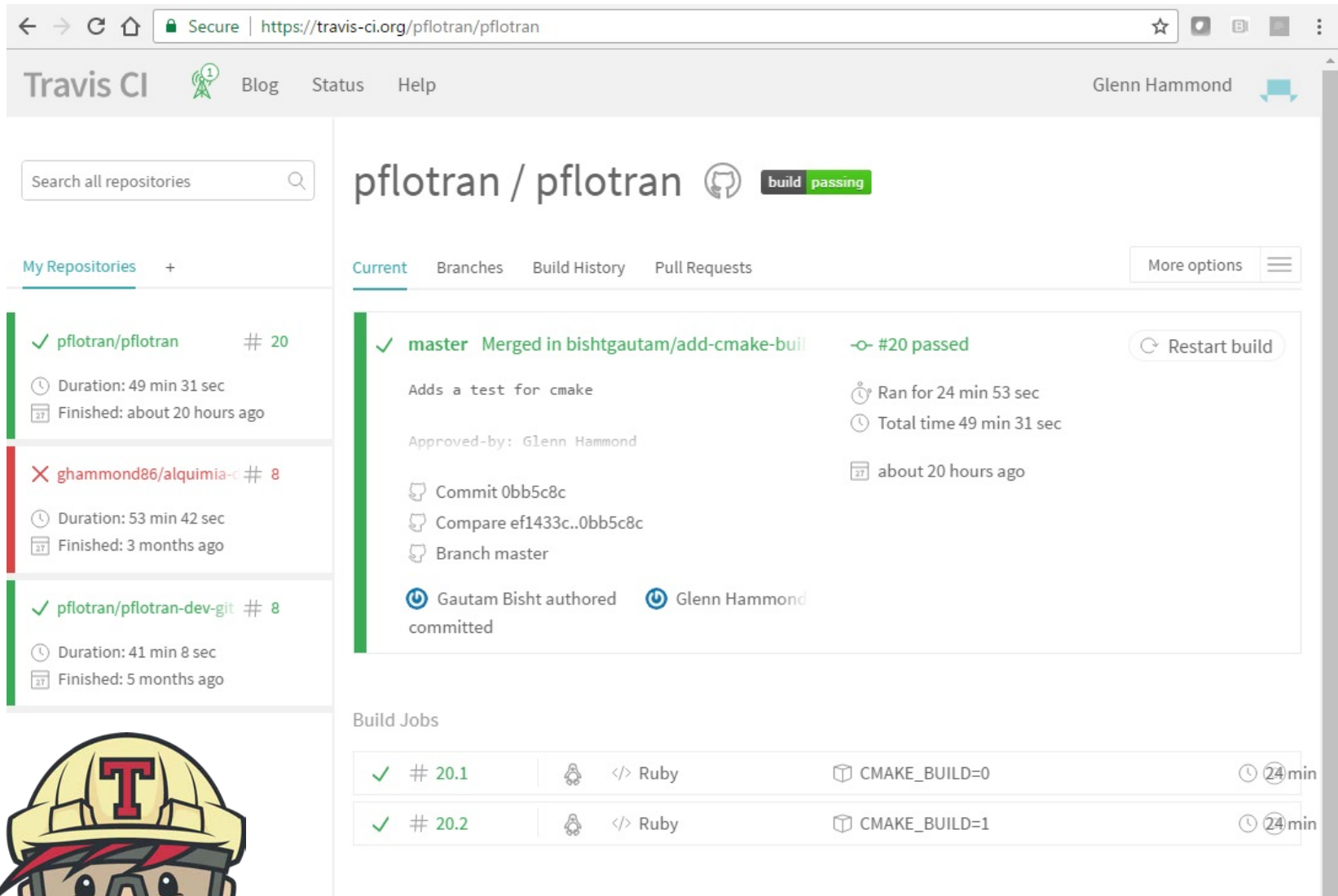
- 2r: 2-norm of residual
- 2x: 2-norm of current solution
- 2u: 2-norm of update
- ir: inf-norm of residual
- iu: inf-norm of update
- rsn: converged reason (corresponding integer in brackets)
  - 0: iterating (not converged)
  - atol[2]:  $2r < ATOL$
  - rtol[3]:  $2r < RTOL * 2r_{initial}$
  - stol[4]:  $2u < STOL * 2x$
  - itol\_res[10]:  $ir < ITOL\_RES$
  - itol\_upd[11]:  $iu < ITOL\_UPDATE$

# PFLOTRAN Bitbucket Site



The screenshot shows a web browser window displaying the PFLOTRAN Bitbucket Wiki page. The browser's address bar shows the URL <https://bitbucket.org/pfлотran/pfлотran/wiki/Home>. The page features a blue sidebar on the left with navigation options: Home, Search, and a plus sign. Below these are icons for Overview, Source, Commits, Branches, Pull requests, Pipelines, Issues, Wiki (highlighted), Downloads, Graphs, and Settings. At the bottom of the sidebar are options to 'SHARE YOUR THOUGHTS', 'Give feedback', and 'Turn off new nav'. The main content area shows the breadcrumb 'PFLOTRAN / code / pfлотran' and the title 'Wiki'. There are buttons for 'Create page' and 'Clone wiki'. Below the title is the path 'pfлотran / Home' and buttons for 'View', 'History', and 'Edit'. The main content area contains a large 'PFLOTRAN' logo with a blue triangle and lines. Below the logo is a paragraph describing PFLOTRAN as an open source, state-of-the-art massively parallel subsurface flow and reactive transport code. It mentions the GNU LGPL license, Fortran 2003, and the choice of Fortran over C/C++. Another paragraph describes the parallelization through domain decomposition using the MPI-based PETSc framework with pflотran-dev tracking the git *maint* branch of PETSc available through Bitbucket. A third paragraph invites visitors to visit [pflотran.org](http://pflотran.org) for more information and provides a link to the old wiki. Below this is a 'Travis CI Status' section with a green 'build passing' indicator. The page also has sections for 'Installation Instructions' and 'Documentation', both with links to further information.

# Bitbucket → Github → Travis CI



Travis CI | Blog | Status | Help | Glenn Hammond

Search all repositories

My Repositories +

- ✓ pflotran/pflotran # 20  
Duration: 49 min 31 sec  
Finished: about 20 hours ago
- ✗ ghammond86/alquimia-c # 8  
Duration: 53 min 42 sec  
Finished: 3 months ago
- ✓ pflotran/pflotran-dev-git # 8  
Duration: 41 min 8 sec  
Finished: 5 months ago

## pflotran / pflotran

build passing

Current | Branches | Build History | Pull Requests | More options

✓ master Merged in bishtgautam/add-cmake-build #20 passed Restart build

Adds a test for cmake

Ran for 24 min 53 sec  
Total time 49 min 31 sec  
about 20 hours ago

Approved-by: Glenn Hammond

- Commit 0bb5c8c
- Compare ef1433c..0bb5c8c
- Branch master

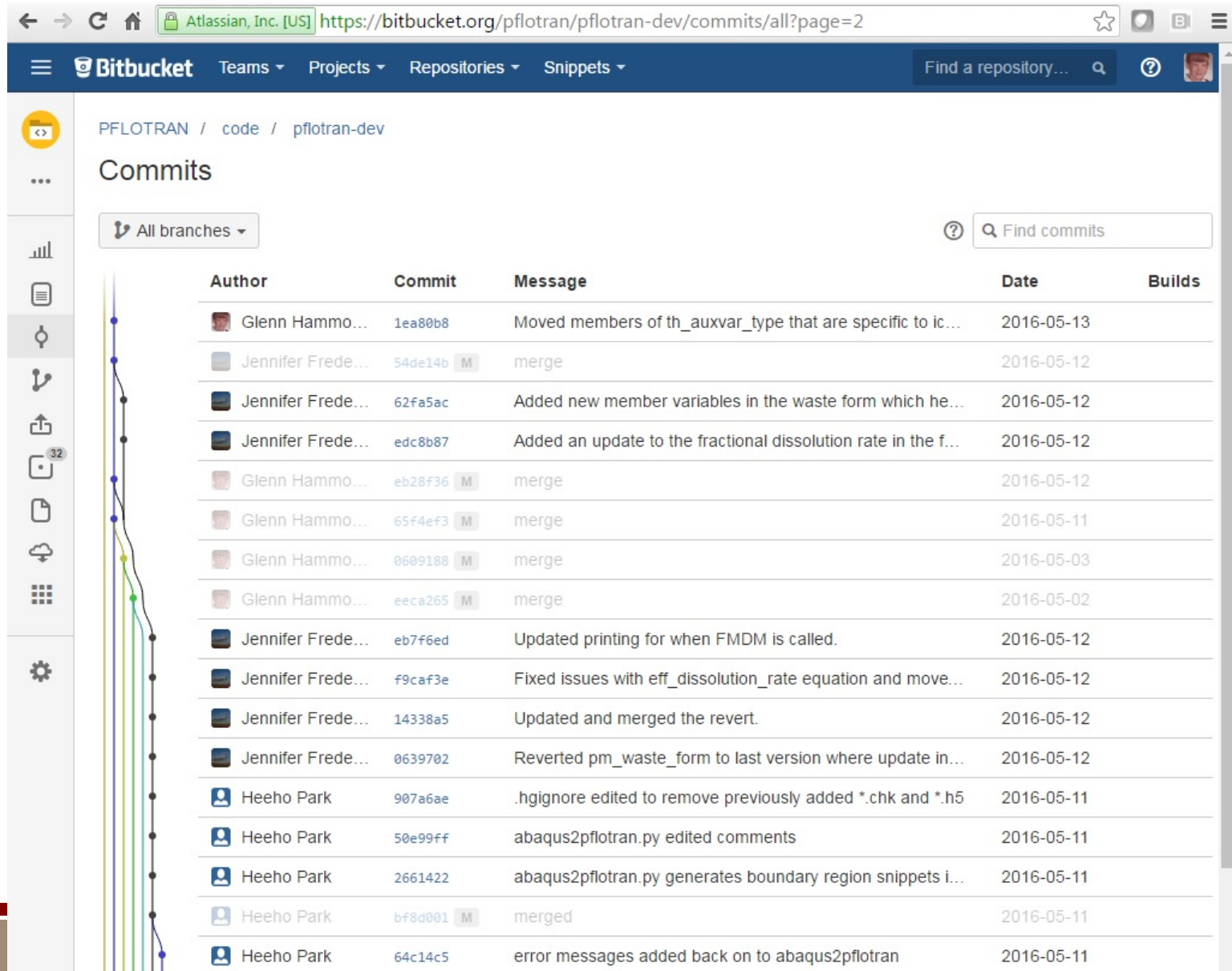
Gautam Bisht authored | Glenn Hammond committed

### Build Jobs

✓ # 20.1	</> Ruby	CMAKE_BUILD=0	24 min
✓ # 20.2	</> Ruby	CMAKE_BUILD=1	24 min



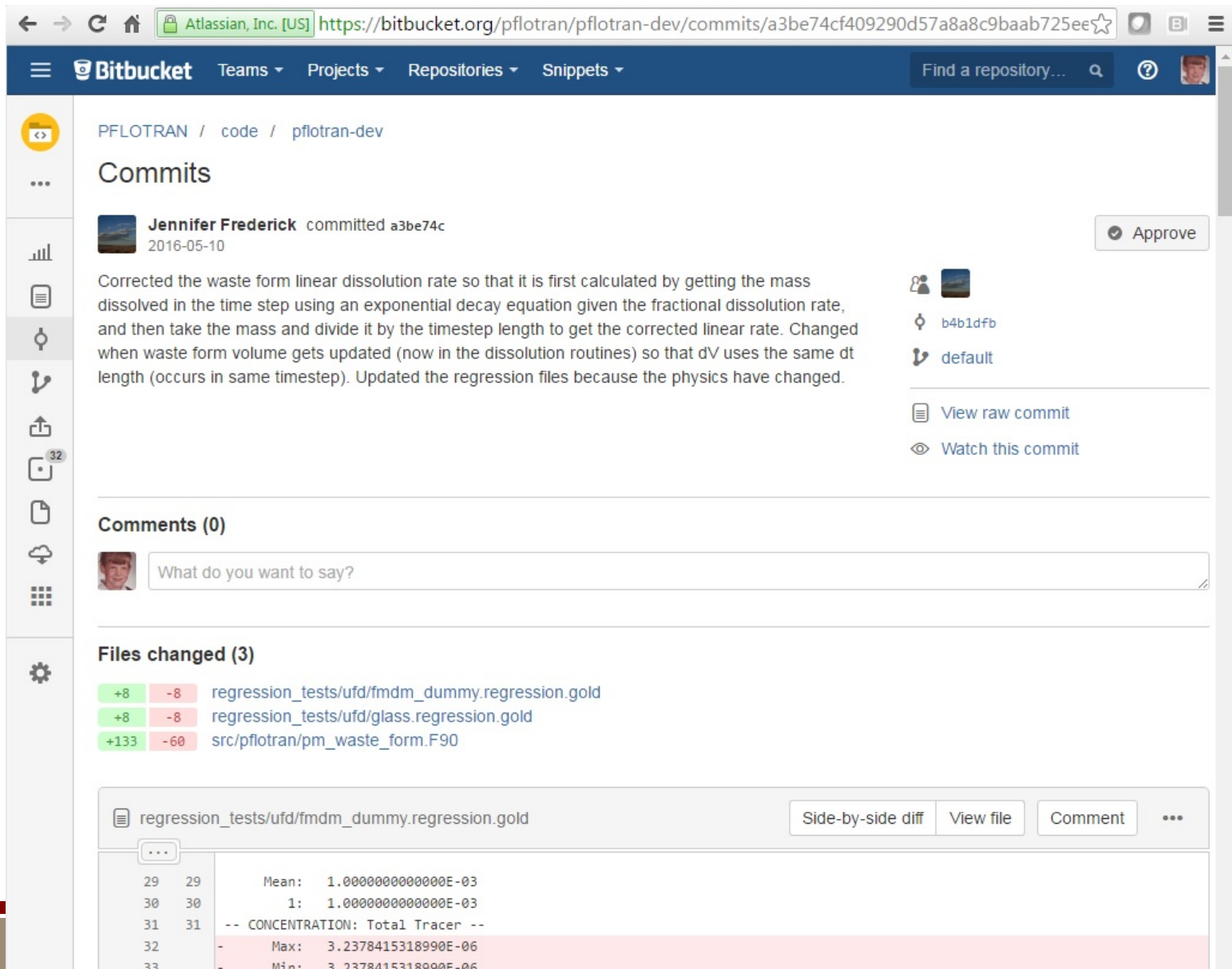
# PFLOTRAN Bitbucket Commit Log



The screenshot shows the Bitbucket web interface for the PFLOTRAN repository. The browser address bar shows the URL: <https://bitbucket.org/pflotran/pflotran-dev/commits/all?page=2>. The page title is "Commits" and it shows a list of 20 commit entries. On the left side, there is a vertical commit history graph showing the progression of branches over time. The commit list includes authors like Glenn Hammo..., Jennifer Frede..., and Heeho Park, with commit messages such as "Moved members of th\_auxvar\_type that are specific to ic...", "merge", "Added new member variables in the waste form which he...", and ".hgignore edited to remove previously added \*.chk and \*.h5".

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...	eeea265 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




← → ↻ 🏠 Atlassian, Inc. [US] <https://bitbucket.org/pflotran/pflotran-dev/commits/a3be74cf409290d57a8a8c9baab725ee> 🔖


☰ **Bitbucket** Teams ▾ Projects ▾ Repositories ▾ Snippets ▾ Find a repository... 🔍 ? 👤

🔗 **PFLOTRAN** / code / pflotran-dev

## Commits


 **Jennifer Frederick** committed a3be74c 2016-05-10 Approve

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

[View raw commit](#)  
[Watch this commit](#)

### Comments (0)

 What do you want to say?

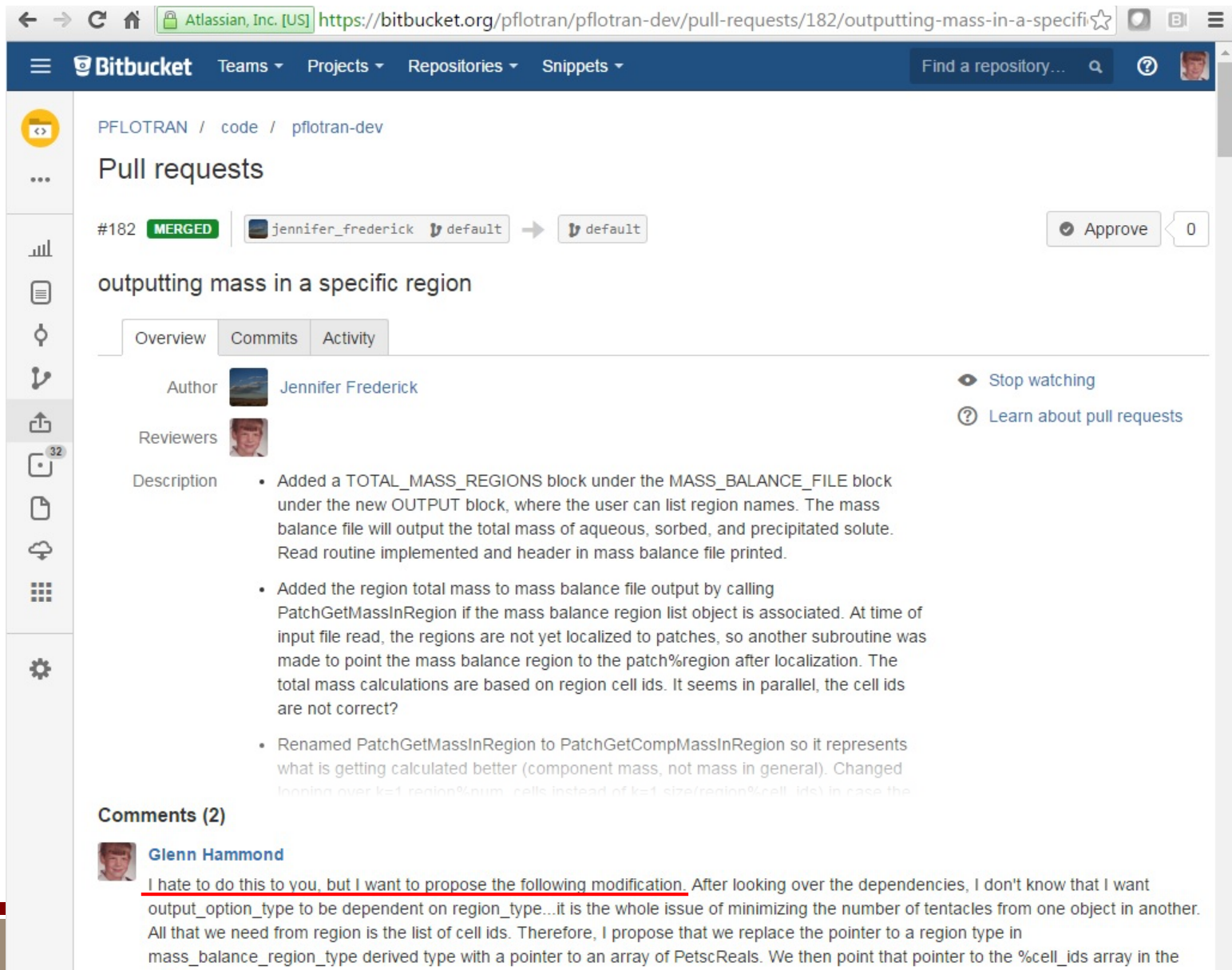
### Files changed (3)

+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 Side-by-side diff View file Comment ⋮

```
29 29      Mean:  1.000000000000E-03
30 30      1:    1.000000000000E-03
31 31  -- CONCENTRATION: Total Tracer --
32 32  -      Max:  3.2378415318990E-06
33 33  -      Min:  3.2378415318990E-06
```

# PFLOTRAN Bitbucket Pull Request



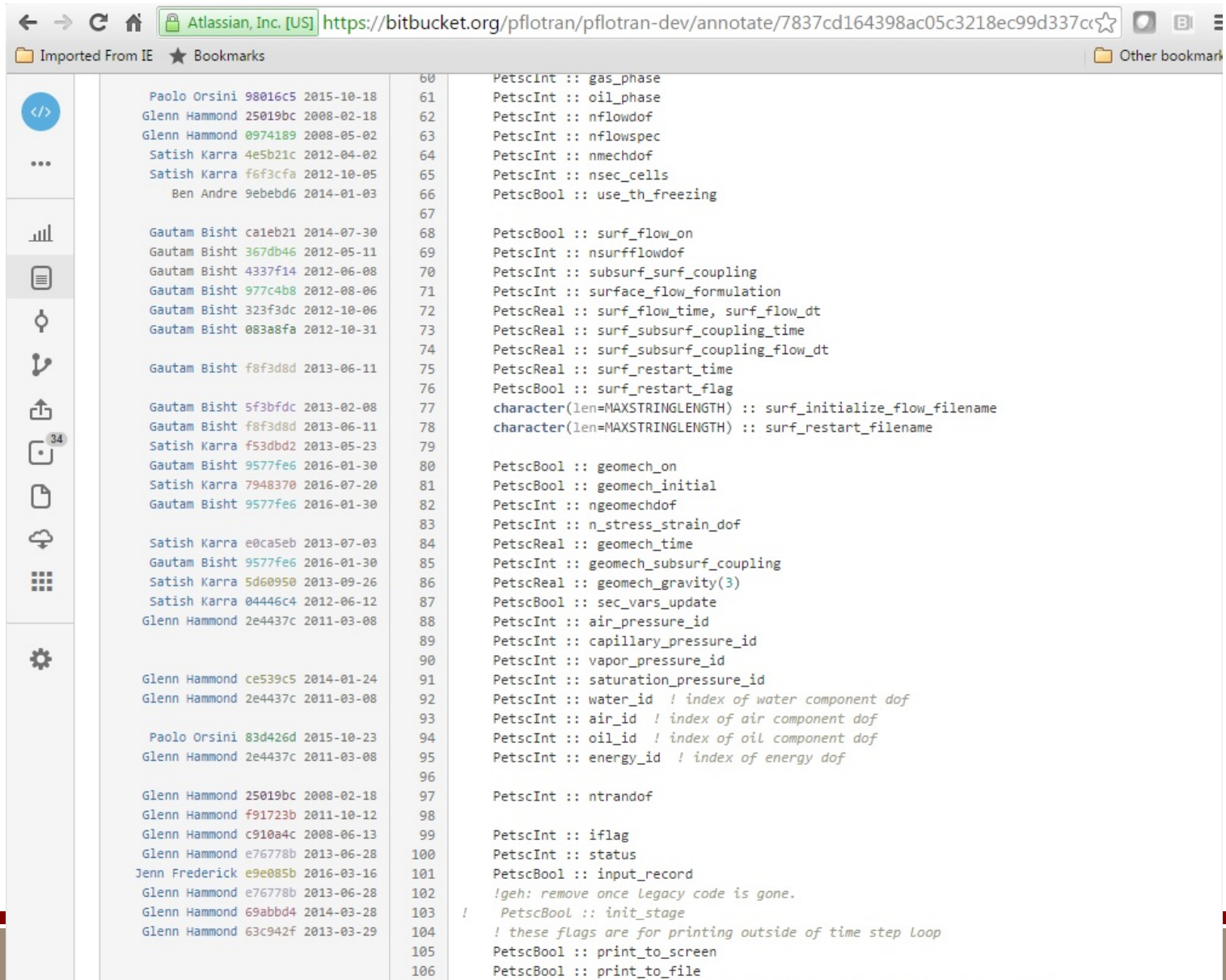
The screenshot shows a Bitbucket pull request page. The browser address bar displays the URL: <https://bitbucket.org/pflotran/pflotran-dev/pull-requests/182/outputting-mass-in-a-specific-region>. The page title is "PFLOTRAN / code / pflotran-dev" and the pull request title is "outputting mass in a specific region". The pull request is #182, is in a "MERGED" state, and is from the "jennifer\_frederick" repository to the "default" branch. There is an "Approve" button with a count of 0. The pull request description includes a list of changes:

- 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 looping over `k=1, region%num_cells` instead of `k=1, size(region%cell_ids)` in case the

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 bookmark


Contributor	Commit Hash	Date	Line	Code
Paolo Orsini	98016c5	2015-10-18	60	PetscInt :: gas_phase
Glenn Hammond	25019bc	2008-02-18	61	PetscInt :: oil_phase
Glenn Hammond	0974189	2008-05-02	62	PetscInt :: nflowdof
Glenn Hammond	0974189	2008-05-02	63	PetscInt :: nflowspec
Satish Karra	4e5b21c	2012-04-02	64	PetscInt :: nmechdof
Satish Karra	f6f3cfa	2012-10-05	65	PetscInt :: nsec_cells
Ben Andre	9ebeb6d	2014-01-03	66	PetscBool :: use_th_freezing
			67	
Gautam Bisht	ca1eb21	2014-07-30	68	PetscBool :: surf_flow_on
Gautam Bisht	367db46	2012-05-11	69	PetscInt :: nsurfflowdof
Gautam Bisht	4337f14	2012-06-08	70	PetscInt :: subsurf_surf_coupling
Gautam Bisht	977c4b8	2012-08-06	71	PetscInt :: surface_flow_formulation
Gautam Bisht	323f3dc	2012-10-06	72	PetscReal :: surf_flow_time, surf_flow_dt
Gautam Bisht	083a8fa	2012-10-31	73	PetscReal :: surf_subsurf_coupling_time
			74	PetscReal :: surf_subsurf_coupling_flow_dt
			75	PetscReal :: surf_restart_time
			76	PetscBool :: surf_restart_flag
Gautam Bisht	5f3bfdc	2013-02-08	77	character(len=MAXSTRINGLENGTH) :: surf_initialize_flow_filename
Gautam Bisht	f8f3d8d	2013-06-11	78	character(len=MAXSTRINGLENGTH) :: surf_restart_filename
Satish Karra	f53dbd2	2013-05-23	79	
Gautam Bisht	9577fe6	2016-01-30	80	PetscBool :: geomech_on
Satish Karra	7948370	2016-07-20	81	PetscBool :: geomech_initial
Gautam Bisht	9577fe6	2016-01-30	82	PetscInt :: ngeomechdof
			83	PetscInt :: n_stress_strain_dof
Satish Karra	e0ca5eb	2013-07-03	84	PetscReal :: geomech_time
Gautam Bisht	9577fe6	2016-01-30	85	PetscInt :: geomech_subsurf_coupling
Satish Karra	5d60950	2013-09-26	86	PetscReal :: geomech_gravity(3)
Satish Karra	04446c4	2012-06-12	87	PetscBool :: sec_vars_update
Glenn Hammond	2e4437c	2011-03-08	88	PetscInt :: air_pressure_id
			89	PetscInt :: capillary_pressure_id
			90	PetscInt :: vapor_pressure_id
Glenn Hammond	ce539c5	2014-01-24	91	PetscInt :: saturation_pressure_id
Glenn Hammond	2e4437c	2011-03-08	92	PetscInt :: water_id ! index of water component dof
			93	PetscInt :: air_id ! index of air component dof
Paolo Orsini	83d426d	2015-10-23	94	PetscInt :: oil_id ! index of oil component dof
Glenn Hammond	2e4437c	2011-03-08	95	PetscInt :: energy_id ! index of energy dof
			96	
Glenn Hammond	25019bc	2008-02-18	97	PetscInt :: ntrandof
Glenn Hammond	f91723b	2011-10-12	98	
Glenn Hammond	c910a4c	2008-06-13	99	PetscInt :: iflag
Glenn Hammond	e76778b	2013-06-28	100	PetscInt :: status
Jenn Frederick	e9e085b	2016-03-16	101	PetscBool :: input_record
Glenn Hammond	e76778b	2013-06-28	102	!geh: remove once legacy code is gone.
Glenn Hammond	69abb4d	2014-03-28	103	! PetscBool :: init_stage
Glenn Hammond	63c942f	2013-03-29	104	! these flags are for printing outside of time step loop
			105	PetscBool :: print_to_screen
			106	PetscBool :: print_to_file

# PFLOTRAN User Support

← → ↻ 🏠 <https://groups.google.com/forum/#!forum/pflotran-users> ☆ 🔔 📧 ☰

**Google** Search for topics 🔍

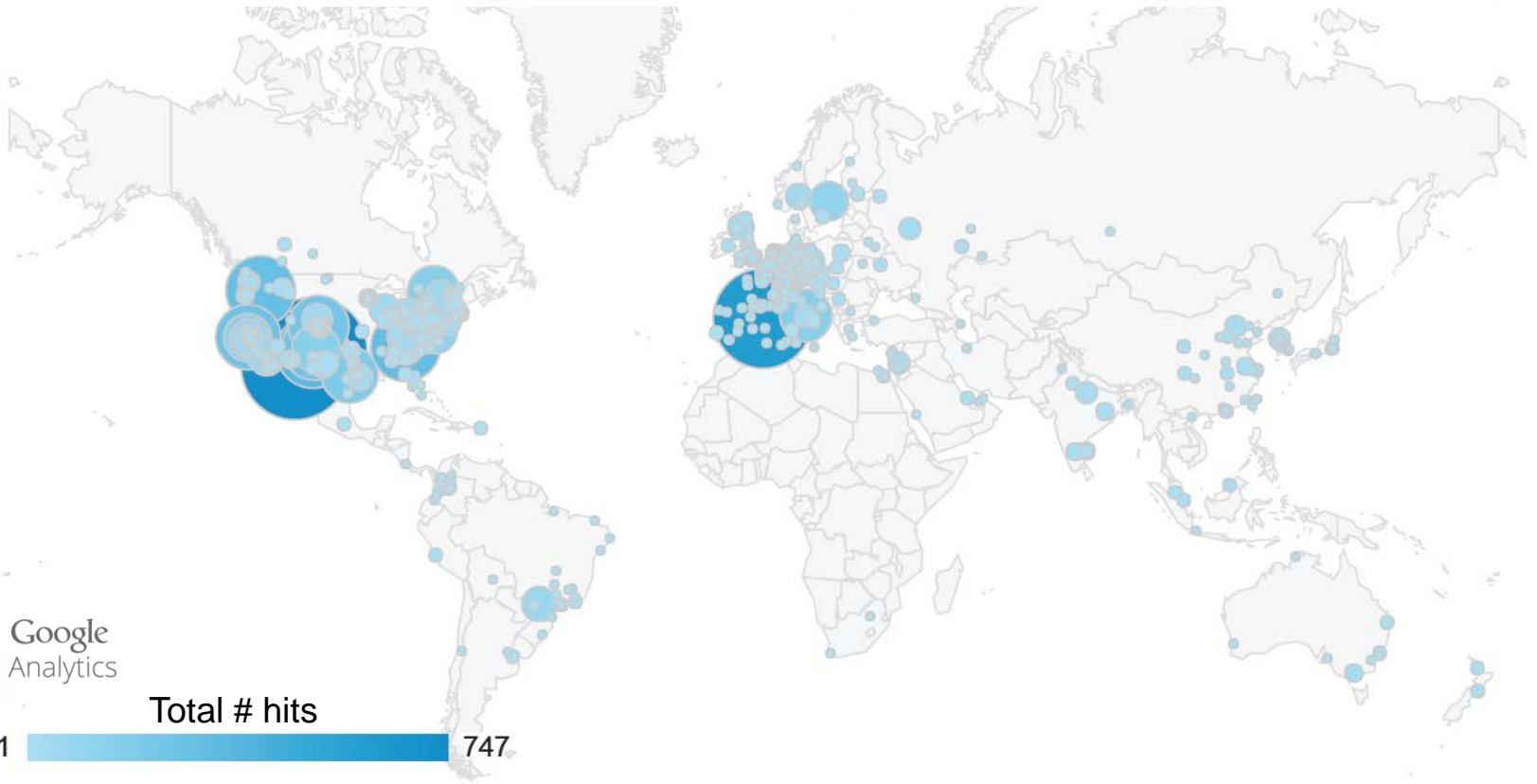
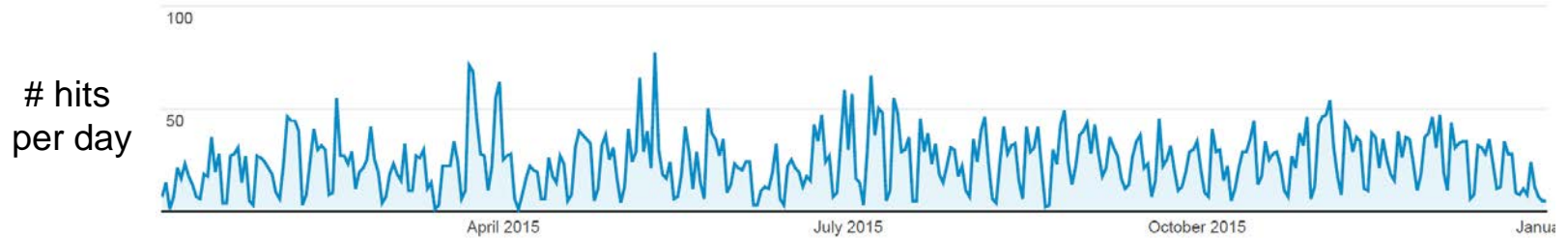
Groups **NEW TOPIC** ↻ Mark all as read Actions ▾ Filters ▾ 👤 ⚙️

▶ **pflotran-users** Shared publicly  
60 of 642 topics (99+ unread) ☆  Manage · Members · About ▾

Welcome to the PFLOTRAN users mailing list.  
[Edit welcome message](#) [Clear welcome message](#)

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

# Hits on PFLOTRAN Bitbucket Site in 2015



# Hits on PFLOTRAN Bitbucket Site in 2016

