1 Introduction

This specification documents features required by PFLOTRAN to simulate physical and chemical processes within the subsurface environment. These required features are divided into functional and non-functional requirements. Functional requirements consist of physical or chemical processes, numerical methods, user interfaces, etc. Non-functional requirements include runtime performance metrics (e.g. scalability, performance), software maintenance and code availability. The specification maps these features to tests designed to verify feature accuracy.

2 Functional Requirements

Fix indentation of lists.

PFLOTRAN 's functional requirements are divided into the following categories:

- Material properties
- Constitutive relations
- Physic and chemistry
- Forcing
- Numerical methods
- Assignment of processes and properties to regions
- User interaction.

2.1 Material Properties

PFLOTRAN shall assign the following soil material properties to each grid cell:

MP 1.Electical conductivity
MP 2.Heat capacity
MP 3.Intrinsic permeability
MP 4.Porosity
MP 5.Soil particle density
MP 6.Thermal conductivity
MP 7.Tortuosity

2.1.1 Assignment of Material Properties

MP 8.PFLOTRAN shall assign each grid cell a material ID.
MP 9.PFLOTRAN shall support non-contiguous material IDs (e.g. 1,2,5).
MP 10.PFLOTRAN shall have the ability to inactivate grid cells based on material ID.
MP 11.PFLOTRAN shall have the ability to assign material properties by material ID.

PFLOTRAN shall assign the following heterogeneous material properties:

- Electical conductivity
- Permeability
- Porosity
- Soil particle density
- Tortuosity

by:

MP 12.Region. Regions are defined by points, lines, planes, volumes or lists of cell IDs.
MP 13.XYZ coordinate. This approach shall assign a value based on interpolation within a regularly spaced dataset.
MP 14.Cell ID. This approach assigns an externally generated dataset on a cell by cell basis.

2.2 Constitutive Relations

Constitutive relations employ mathematical equations to approximate the observed physical response of a material or fluid under conditions of interest. For instance, an equation of state calculates a fluid density as a function of input parameters pressure and temperature.

2.2.1 Equations of State

PFLOTRAN shall incorporate the following equations of state:

CR 1. PFLOTRAN shall implement the IFC67 (International Formulation Committee, 1967) equations of state for calculating the density, enthalpy, and viscosity of water as a function of pressure and temperature.

CR 2. PFLOTRAN shall implement the IF97 from the International Association for the Properties of Water and Steam (Wagner et al, 2000) equations of state for calculating the density and enthalpy of water and steam as a function of pressure and temperature.

CR 3. PFLOTRAN shall implement an equation of state for water that holds the density, viscosity and enthalpy of water and steam constant regardless of pressure or temperature.

2.2.2 Saturation Functions

In variably saturated or multiphase flow in porous media, it is essential to establish a relation between capillary pressure and saturation. A saturation function calculates the liquid or gas saturation as a function of capillary pressure and vice versa. PFLOTRAN shall incorporate the following and saturation functions:

CR 4. PFLOTRAN shall implement the van Genuchten function (van Genuchten, 1980) to calculate the saturation on a cell given its capillary pressure value.

CR 5. PFLOTRAN shall implement the Brooks-Corey function (Brooks and Corey, 1964) to calculate the saturation on a cell given its capillary pressure value.

2.2.3 Relative Permeability Functions

Relative permeability functions establish a relationship between fluid phase's relative permeability, which ranges between 0 and 1, and its saturation. PFLOTRAN shall incorporate the following relative permeability functions:

CR 6. Mualem-van Genuchten

CR 7. Mualem-Brooks-Corey

CR 8. Burdine-van Genuchten

CR 9. Burdine-Brooks-Corey.

2.2.4 Soil Compressibility Functions

The volume of fluid stored in porous media can change due to pore compaction and fluid expansion. While compaction (or expansion) of fluids is governed by the fluid compressibility (e.g., through an equation of state), change in pore volume is set forth by soil compressibility functions, which relate porosity change to a shift in pressure. PFLOTRAN shall implement the following soil compressibility functions:

CR 10. Leijnse function (Leijnse 1992)

CR 11. An Exponential relationship.

2.2.5 Constitutive Relation Coupling by Region

CR 12. PFLOTRAN shall allow for capillary pressure/saturation functions, relative permeability functions and soil compressibility functions to be specified by region. (HIGH - easy)

2.3 Physics and Chemistry

PFLOTRAN employs mathematical representations of physical and chemical process models to simulate phenomenon in the subsurface. These process models include:

2.3.1 Fluid Flow

PC 1. PFLOTRAN shall simulate single phase variably–saturated flow based on the Richards equation [?].

2.3.2 Multicomponent Solute Transport

PC 2. PFLOTRAN shall simulate solute advection.
PC 3. PFLOTRAN shall simulate solute molecular diffusion.
PC 4. PFLOTRAN shall simulate solute mechanical dispersion (through a diagonal dispersion tensor).

2.3.3 Biogeochemical Reaction

PC 5. PFLOTRAN shall simulate sorption using a linear isotherm (K_D).
PC 6. PFLOTRAN shall simulate first-order radioactive decay in the aqueous and sorbed phases.

2.3.4 Geophysics

PC 7. PFLOTRAN shall simulate electrical resistivity tomography (ERT).

2.4 Forcing Requirements (Boundary Conditions and Source/Sinks)

PFLOTRAN shall allow the specification of the following flow boundary conditions and source/sinks:

2.4.1 Fluid Flow

Initial Conditions

FR 1. PFLOTRAN shall assign initial pressures to cells.
FR 2. PFLOTRAN shall map initial pressures to cells from a hydrostatic fluid column. The column shall be calculated as a function of a pressure defined at a reference surface, the vertical distance from the reference surface, fluid density and gravity. Note that fluid density shall not be constant when calculated using an equation of state other than CONSTANT.

Dirichlet Boundary Conditions

FR 3. PFLOTRAN shall assign boundary pressures to boundary cell faces.
FR 4. PFLOTRAN shall map boundary pressures to cell faces from a hydrostatic fluid column. The column shall be calculated as a function of a pressure defined at a reference surface, the vertical distance from the reference surface, fluid density and gravity. Note that fluid density shall not be constant when calculated using an equation of state other than CONSTANT.
FR 5. PFLOTRAN shall assume a no flow (zero flux) condition when no boundary condition is assigned to a boundary face.

Neumann Bouondary Conditions

FR 6. PFLOTRAN shall prescribe Darcy fluxes across boundary cell faces.

Source/Sinks

FR 7. PFLOTRAN shall add/remove fluid mass from cells using a mass rate equation.
FR 8. PFLOTRAN shall add/remove fluid mass from cells in a region using a mass rate equation that is scaled by the ratio of the volume of the cell to the sum of cell volumes in the region.
FR 9. PFLOTRAN shall add/remove fluid mass from cells in a region using a mass rate equation that is scaled by the ratio of the product of cell volume and intrinsic permeability to the sum of cell volumes and permeabilities in the region.
FR 10. PFLOTRAN shall add/remove fluid mass from cells using a volumetric rate equation.
FR 11. PFLOTRAN shall add/remove fluid mass from cells in a region using a volumetric rate equation that is scaled by the ratio of the volume of the cell to the sum of cell volumes in the region.

FR 12.PFLOTRAN shall add/remove fluid mass from cells in a region using a volumetric rate equation that is scaled by the ratio of the product of cell volume and intrinsic permeability to the sum of cell volumes and permeabilities in the region.

General

FR 13.PFLOTRAN shall support time-varying boundary conditions and source/sinks.

2.4.2 Representation of Initial Conditions

PFLOTRAN shall assign heterogeneous initial conditions across the domain by:

RIC 1.Region. Regions are defined by points, lines, planes, volumes or lists of cell IDs.

RIC 2.XYZ coordinate. This approach shall assign a value based on interpolation within a regularly spaced dataset.

RIC 3.Cell ID. This approach assigns an externally generated dataset on a cell by cell basis.

HIGH - easy

2.4.3 Representation of Boundary Conditions and Source/Sink terms

PFLOTRAN shall allow the specification of various boundary conditions and source-sink terms using the following strategies:

RBC 1.Region. Regions are defined by points, lines, planes, volumes or lists of cell IDs.

RBC 2.XYZ coordinate. This approach shall assign a value based on interpolation within a regularly spaced dataset.

2.5 Numerical Methods

2.5.1 Finite volume implementation

NM 1.PFLOTRAN shall employ the cell-centered finite volume method with the two-point flux approximation to discretize the governing mass and heat conservation equations.

2.5.2 Time Stepping

NM 2.PFLOTRAN shall have the ability to vary the time step size. The time stepping will depend on an initial time step size, a minimum and maximum time step size, and a maximum growth and reduction factor. The maximum time step size shall have the ability to change during the simulation time. Time step size shall increase or decrease according to growth and reduction factor as a function of the number of iterations needed for convergence. HIGH - medium

NM 3.PFLOTRAN shall have the ability to restrict time steps as a function of the maximum flow velocity and grid discretization such that CFL (Courant–Friedrichs–Lewy) number is not exceeded. HIGH - easy

NM 4.PFLOTRAN shall allow different process models, such as flow and transport, to comply with different time step settings. MEDIUM - easy

2.5.3 Nonlinear solvers

NM 5.PFLOTRAN shall implement a Newton-Raphson strategy to solve the set of nonlinear governing equations and iteratively drive the norm of the residual vector to below a desired convergence tolerance. LOW - easy

NM 6.PFLOTRAN shall report convergence failure and cut the timesetp size if the maximum number of Newton iterations is reached. LOW-easy

PFLOTRAN shall support the following Newton-Raphson convergence criteria:

NM 7.Convergence is met when the 2-norm of residual is less than ATOL.

NM 8.Convergence is met when the 2-norm of residual is less than RTOL multiplied by the 2-norm of the residual from the first Newton iteration.

NM 9. Convergence is met when the 2-norm of the update (difference between the current and the previous iteration solution) is less than STOL multiplied by the 2-norm of the previous iteration solution.
NM 10. Convergence is met when the infinity norm of the Newton update (difference between the current and previous solution) is less than ITOL_UPDATE.
NM 11. The solver diverges (fails) when the 2-norm of the residual for the current Newton iteration is greater than DTOL multiplied by the 2-norm of the relative from the first iteration.
MEDIUM - easy

2.5.4 Linear solvers

The Newton Raphson method iteratively solves linear systems of equations for updates to the solution.

PFLOTRAN shall support the following linear system solvers:
NM 12. Direct LU Decomposition
NM 13. Iterative BiCGSTAB and GMRES Krylov solvers

PFLOTRAN shall implement the following Krylov solver convergence criteria:
NM 14. Convergence is met when the 2-norm of residual is less than ATOL.
NM 15. Convergence is met when the 2-norm of residual is less than RTOL multiplied by the 2-norm of the residual from the first linear iteration.
NM 16. The solver fails once a prescribed maximum number of iterations (MAXIT) is met.
NM 17. The solver diverges (fails) when the 2-norm of the residual for the current linear iteration is greater than DTOL multiplied by the 2-norm of the relative from the first iteration.
MEDIUM - easy

2.5.5 Gridding

PFLOTRAN shall support the following types of grids:
NM 18. Structured grids: Cartesian grids of hexahedron cells with variable grid spacing.
NM 19. Implicit unstructured grids: Unstructured grids defined by vertices (x,y,z coordinates) and cells (lists of vertices). This approach is similar to finite element grids composed of nodes and elements. Implicit unstructured grids shall support the following types of cells: tetrahedron (4 vertices), pyramid (5 vertices), wedge (6 vertices), and hexahedron (8 vertices). Grid cells are defined by a list of vertex ids and vertices are defined by a coordinate.
NM 20. Explicitly unstructured grids: Unstructured grids defined by volumes and connectivity (between volumes). Cells are defined by an id, the cell-center coordinates, and the cell volume. Connectivity between two cells is composed by the id for each cell, the area that connects the cells and the face-center coordinates.

NM 21. PFLOTRAN shall allow for a non-zero structured grid origin (i.e. origin $!= < 0, 0, 0 >$).

2.6 User Interaction

2.6.1 Input format

UI 1. PFLOTRAN shall read an ASCII file as input. The input file shall be divided into blocks and sub-blocks. The block that specifies the type of simulation to run (e.g.: subsurface) and the process model to use (e.g.: subsurface flow or subsurface transport) shall be called the simulation block. The simulation block shall be required in every input file. The remaining blocks shall define numerical methods, solver options, domain discretization, material properties, constitutive relations, time step options, output options, initial and boundary conditions and regions within the domain. LOW - easy
UI 2. PFLOTRAN shall allow for the definition of regions within the domain, which can be cuboids, rectangles or points. HIGH - easy
UI 3. Except for *Celsius* for temperature and *bars* for gas partial pressures in chemical constraints, PFLOTRAN shall default to SI units when units are not specified in the input file. LOW - hard
UI 4. PFLOTRAN shall use default units of molarity [moles per liter or M] for concentration.

UI 5. PFLOTRAN shall print a descriptive error message to the screen and terminate simulation execution when any required keyword, block, sub-block or property is missing. LOW - hard

2.6.2 Output format

UI 6. PFLOTRAN shall print simulator performance metrics (e.g. time step number, time, time step size, norms calculated for convergence calculations, solver iteration numbers, time step cuts, etc.) to the screen for specific time steps. The user may specify a frequency for printing such data to the screen, or they may deactivate the printing. LOW - easy

UI 7. PFLOTRAN shall write state variables to files at specific times. HIGH - easy

PFLOTRAN shall support the following output file formats:

UI 8. Snapshot file: A snapshot file shall store the value of specific state variables over the entire domain at a specific time. PFLOTRAN shall support the following formats: Tecplot block, Tecplot point and HDF5. HIGH - easy

UI 9. Observation file: An observation file shall store the value of specific state variables at specified cells throughout time. The observation file shall have a column-delimited ASCII format. HIGH - easy

UI 10. Mass balance file: A mass balance file shall store the global mass balance and instantaneous and cumulative fluxes at all boundaries and source/sinks for fluids and solutes throughout time. The mass balance file shall have a column-delimited ASCII format. HIGH - easy

UI 11. Checkpoint file: A binary file that stores all information necessary to restart a simulation at a point in time prior to the final simulation time and match the final simulation results.

3 Non-Functional Requirements

3.1 Runtime Performance

At the end of a simulation, PFLOTRAN shall report:

NFR 1. Total wall clock run time.

NFR 2. Wall clock time spent in each process model.

NFR 3. Total number of linear and nonlinear solver iterations.

NFR 4. Number of time step cuts and wasted solver iterations due to time step cuts.

3.2 Software Maintenance

NFR 5. PFLOTRAN source code shall be tracked using a distributed version control system.

NFR 6. PFLOTRAN source code shall meet the coding standards specified within the PFLOTRAN Developer's guide.

NFR 7. PFLOTRAN bug reporting shall adhere to the approach specified within the PFLOTRAN Developer's guide.

NFR 8. PFLOTRAN source code development branches shall pass automated regression and units testing prior to being merged with the master branch.

NFR 9. PFLOTRAN source code shall be peer reviewed for consistency with the coding standards specified in the PFLOTRAN Developer's guide prior to being merged with the master branch.

3.3 User support

NFR 10. The PFLOTRAN project shall maintain separate user and developer mailing lists to provide users with a means of reporting bugs, discussing questions and supporting community needs.

3.4 Code Availability

NFR 11. PFLOTRAN shall be licensed as open source (GNU LGPL).

NFR 12. PFLOTRAN shall be stored in an online repository with open access (no username/password required).

4 Tests

Test a.Source/sink test: Mass rate
Test b.Source/sink test: Scaled mass rate by volume
Test c.Source/sink test: Scaled mass rate by permeability
Test d.Source/sink test: Volume rate
Test e.Source/sink test: Scaled volume rate by volume
Test f.Source/sink test: Scaled volume rate by permeability
Test g.1D fully saturated hydrostatic initial condition: Problem 2.2.6 from Kolditz et al. 2015
Test h.1D fully saturated Dirichlet BC: Problem 2.2.7 from Kolditz et al. 2015
Test i.1D fully saturated Neumann BC: Problem 2.2.8 from Kolditz et al. 2015
Test j.2D fully saturated Dirichlet and Neumann BC:Problem 2.2.10 from Kolditz et al. 2015
Test k.1D variably saturated from Celia et al. 1990
Test l.2D variably saturated: Infiltration in a large caisson: problem 10.13.3 from Feflow's manual
Test m.EOS test for IFC67: compare density calculations between PFLOTRAN and python script from STOMP documentation
Test n.EOS test for IF97: compare density calculations between PFLOTRAN and online calculator
Test o.Representation of material properties by cell ID on structured grid (with a 2x2x2 cube)
Test p.Representation of material properties by cell ID on structured grid using random correlated fields of porosity and permeability
Test q.Representation of material properties by cell ID on unstructured grid (with a 2x2x2 cube)
Test r.Representation of material properties by location using gridded datasets on structured grid
Test s.Representation of material properties by location using gridded datasets on unstructured grid
Test t.Representation of material properties by location using regions on structured grid
Test u.Representation of material properties by location using regions on unstructured grid
Test v.Representation of material properties with IJK indices on structured grid
Test w.Test capability of inactivating cells
Test x.Test non-contiguous material IDs
Test y.Test the ability to create structured grids with irregular spacing.
Test z.Representation of initial conditions by cell ID on structured grid (with a 2x2x2 cube)
Test α.Representation of initial conditions by cell ID on unstructured grid (with a 2x2x2 cube)
Test β.Representation of initial conditions by cell ID on structured grid using random correlated fields of porosity, permeability, and initial pressure.
Test γ.Representation of initial conditions by location using regions on structured grid
Test δ.Representation of initial conditions by location using regions on unstructured grid
Test ε.Representation of initial conditions with IJK indices on structured grid
Test ζ.Representation of initial conditions by location using gridded datasets on structured grid
Test η.Representation of boundary conditions by location using gridded datasets on structured grid (short-course example and comparison available at www.pflotran.org/qa)
Test θ.Test time step variablity following the growth factor until it reaches the maximum time step size.
Test ι.Test time step variablity following the reduction factor until it reaches the maximum number of consecutive cuts.
Test κ.Test time step variablity following the reduction factor until it reaches the minimum time step size.
Test λ.`https://groups.google.com/g/pflotran-users`
Test μ.`https://groups.google.com/g/pflotran-dev`

4.1 Test matrix for Constitutive Relations

Requirement	Tests that use capability	Tests that verify capability
CR 1	Test m	Yes
CR 2	Test n	Yes
CR 3	?	Yes
CR 4	Test k, Test l, Test o, doc-dev	Yes
CR 5	doc-dev	Yes
??	Test k, Test l, Test o, doc-dev	Yes
??	doc-dev	Yes
CR 10	Test l, Test o	No
CR 11	**??**	No
CR 12	**??**	No

4.2 Test matrix for Physics and Chemistry

Requirement	Tests that use capability	Tests that verify capability
PC 1	Test a to Test x	Yes
??		
PC 5		
PC 6		
PC 7		

4.3 Test matrix for Flow Initial Conditions

Requirement	Tests that use capability	Tests that verify capabil
FR 1		
FR 2		
Start here. Sort out why the ICXXX is duplicated. **??**		
??		
??		
??		
??		
??		

4.4 Test matrix for Flow Forcing Requirements

Requirement	Tests that use capability	Tests that verify capability
FR 3	Test h, Test j, Test k	Yes
FR 4	Test g, Test w	Yes
FR 6	Test i, Test l	Yes
FR 7	Test a	Yes
FR 8	Test b	Yes
FR 9	Test c	Yes
FR 10	Test d	Yes
FR 11	Test e	Yes
FR 12	Test f	Yes
FR 13	Test i	Yes
FR 5	Test l, Test o	Yes

4.5 Test matrix for Numerical Methods

Requirement	Tests that use capability	Tests that verify capability
NM 1	??	
NM 2	Test θ, Test ι, Test κ	Yes
NM 3	??	
NM 4	??	
NM 5	??	
NM 6	??	
NM 7	??	
NM 8	??	
NM 9	??	
NM 10	??	
NM 11	??	
NM 12		
NM 13		
NM 14	??	
NM 15	??	
NM 16	??	
NM 17	??	
NM 18	Test g to Test l, Test y	Yes
NM 19	Test q, Test s	Yes
NM 20	Test b, Test c, Test e, Test f	Yes

4.6 Test matrix for Representation of Material Properties

Requirement	Tests that use capability	Tests that verify capability
MP 12	Test t, Test u, Test v	Yes
MP 13	Test r, Test s	Yes
MP 14	Test o, Test q	Yes

4.7 Test matrix for Representation of Initial Conditions

Requirement	Tests that use capability	Tests that verify capability
RIC 1	Test γ, Test δ, Test ε	Yes
RIC 2	Test ζ	Yes
RIC 3	Test z, Test β, Test α,	Yes

4.8 Test matrix for Representation of Boundary Conditions and Source/Sink terms

Requirement	Tests that use capability	Tests that verify capability
RBC 1	Test b, Test c, Test e, Test f	Yes
RBC 2	Test η	Yes

4.9 Test matrix for User Interaction

Requirement	Tests that use capability	Tests that verify capability
UI 1	??	
UI 2	??	
UI 3	??	
UI 5	??	
UI 7	??	
UI 6	??	
??	??	
UI 8	??	
UI 9	??	
UI 10	??	

4.10 Test matrix for Non-Functional Requirements

Requirement	Tests that use capability	Tests that verify capability
NFR 5	??	
NFR 6	??	
NFR 7	??	
NFR 8	??	
??	??	
NFR 10	??	
??	??	