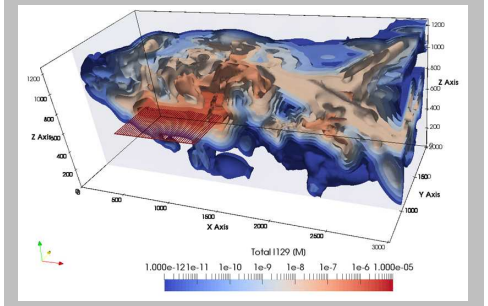
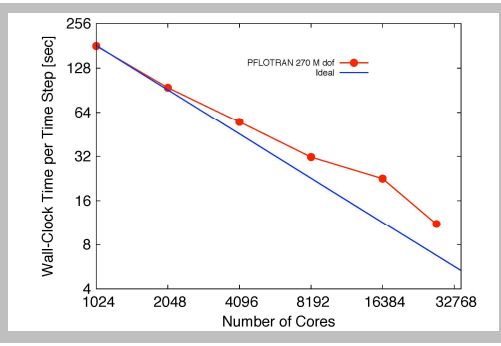
SAND2017-9428PE

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





PFLOTRAN Overview

for the French National Radioactive Waste Management Agency

Glenn E. Hammond

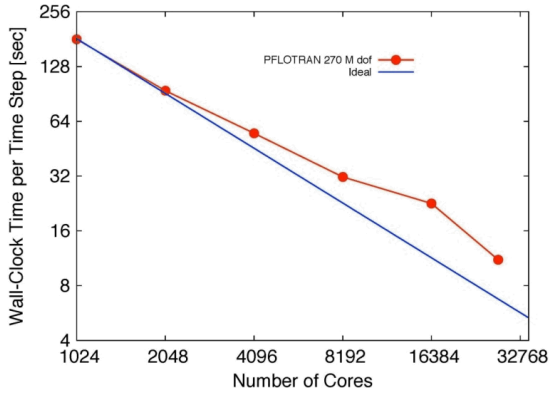
September 19, 2017

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

Sandia National Laboratories

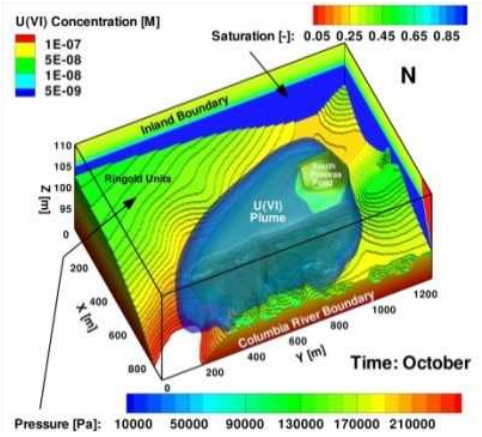
- Petascale reactive multiphase flow and transport code
- Open source license (GNU LGPL 2.0)
- Object-oriented Fortran 2003/2008
  - Pointers to procedures
  - Classes (extendable derived types with member procedures)



- 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

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

Discretization and Numerical Methods

- Spatial discretization
  - Finite volume (2-point flux default)
  - Structured and unstructured grids
- Time discretization: fully-implicit 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

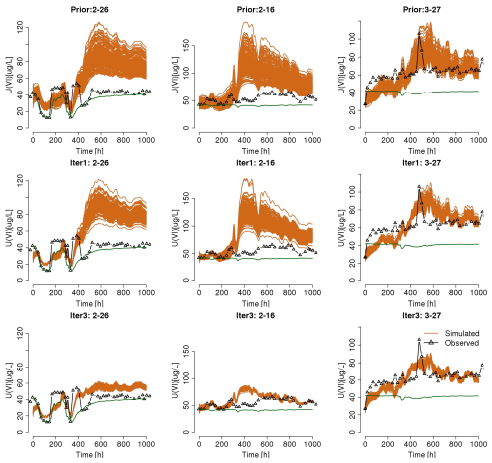
Deep Borehole Waste Disposal

Emily Stein, SNL, 2015

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



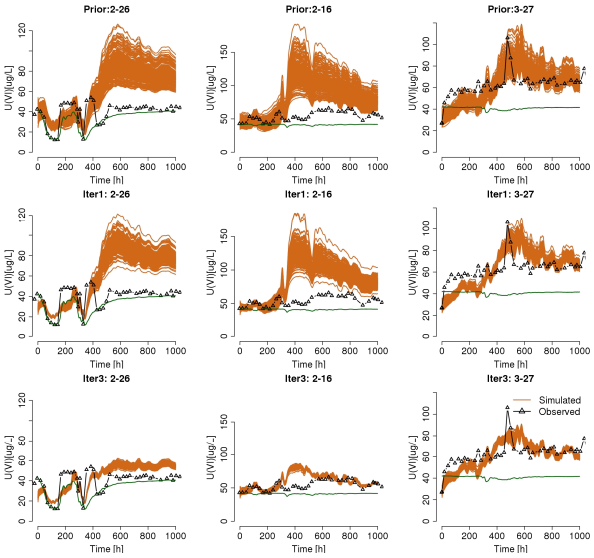
Xingyuan Chen, PNNL, 2011

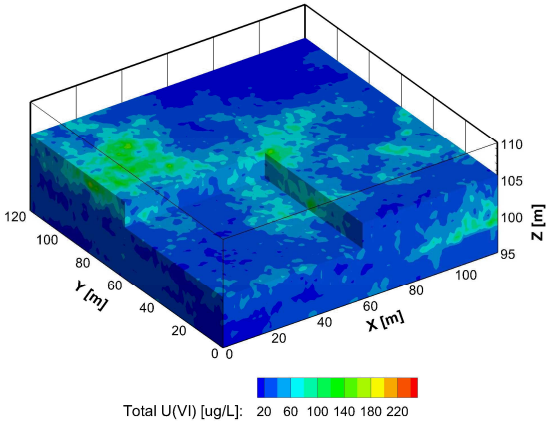
Benefits of Massively Parallel HPC

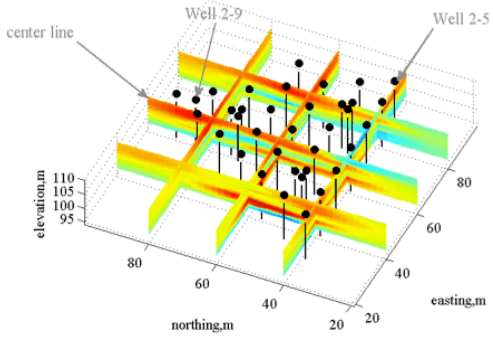
Sandia National Laboratories

Data Assimilation at the Hanford 300 Area (Xingyuan Chen, 2011)

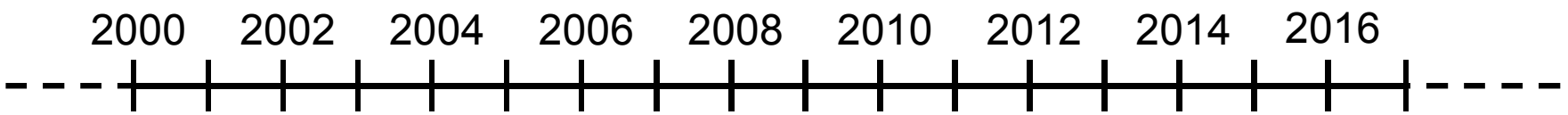
- Problem domain:
  - 120×120×15m
  - $\Delta x, \Delta y, \Delta z$ = 1,1,0.5 m
  - 432K grid cells
  - 15 chemical species
  - 6.48M dofs total
- 1-2 month simulation:
  - Δt = 1 hour
- Computing, e.g.
  - 128 cores (single realization)
  - 64,000 cores (500 realizations)
  - 1 hour wallclock runtime
  - ~7 cpu years







PFLOTRAN Development Timeline



Peter Lichtner

Glenn Hammond

Richard Mills

Chuan Lu

Jitu Kumar

Gautam Bisht

Satish Karra

Ben Andre

Nate Collier

Heeho Park

Paolo Orsini

Jennifer Frederick

Ramesh Sarathi

First release

SciDAC-funded rewrite

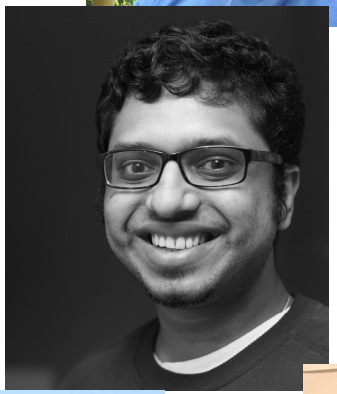
Process model refactor

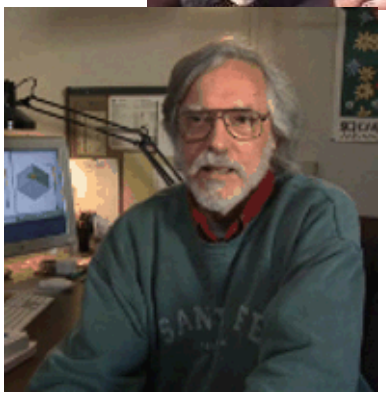
PFLOTRAN Developers

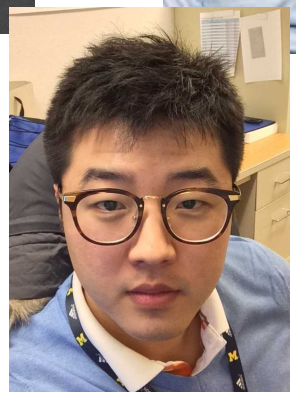
Sandia National Laboratories





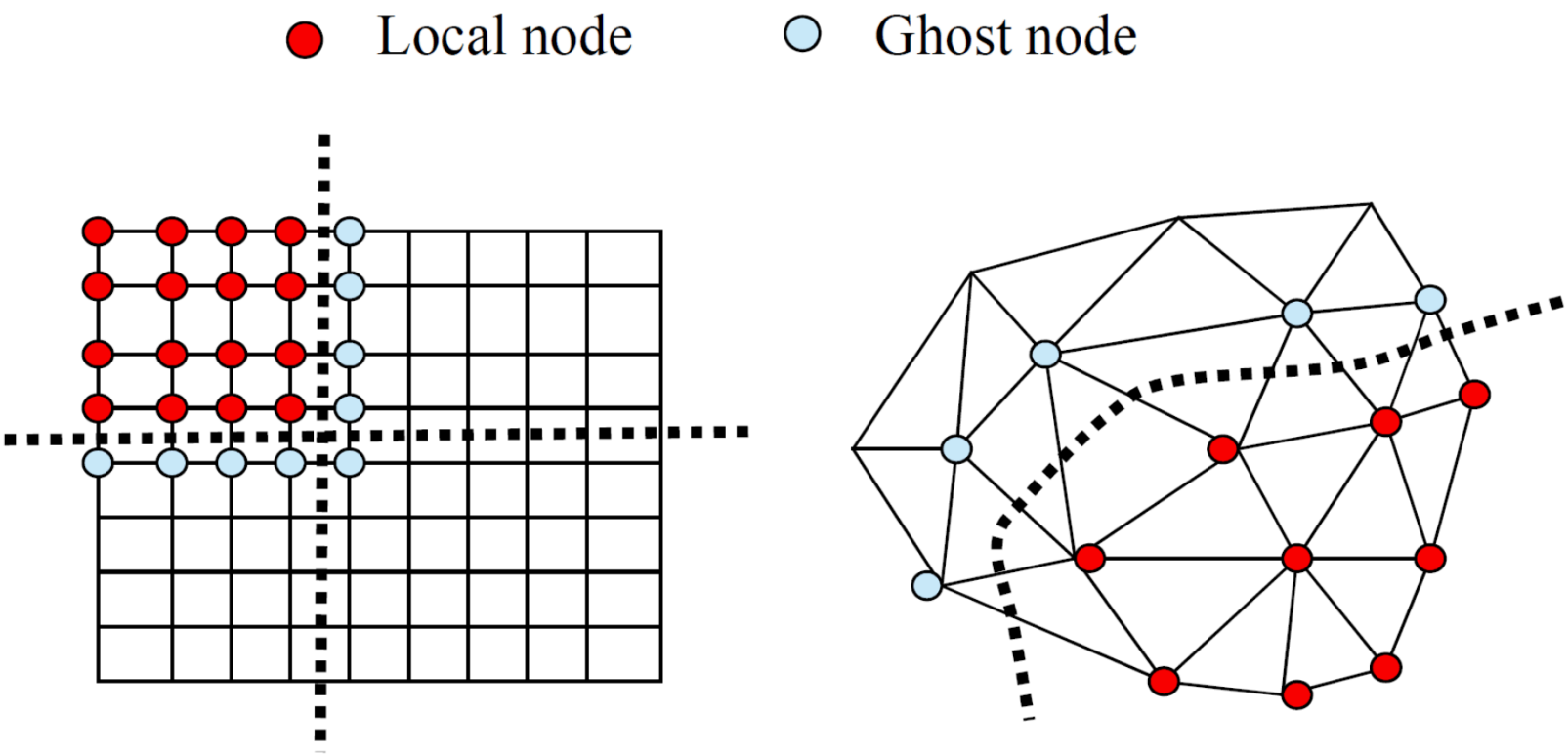


Approach to Parallelization

Sandia National Laboratories

Domain Decomposition

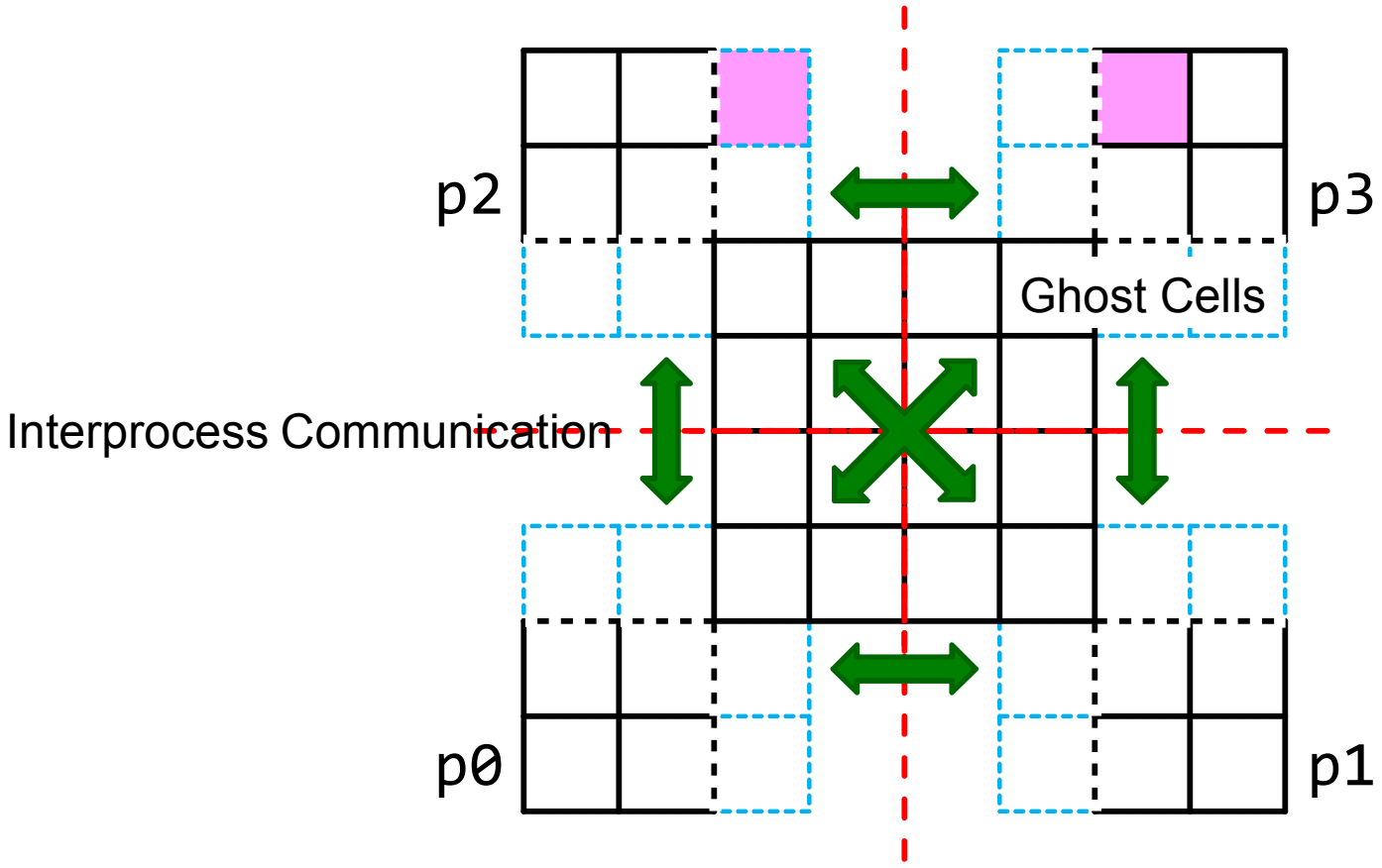


Ghost node information used only in flux calculations.

Domain Decomposition

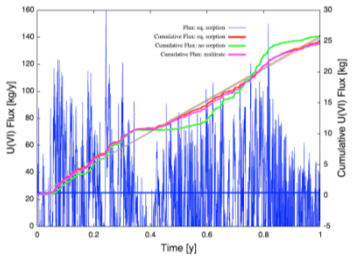
Sandia National Laboratories

Structured Grid Example

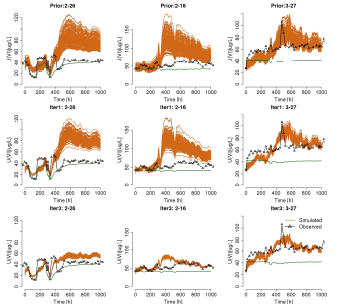


PFLOTRAN Output Formats

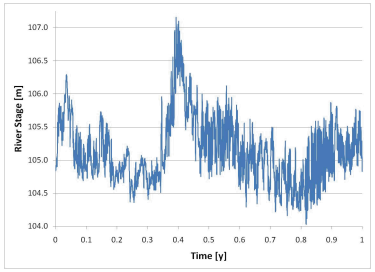
- Excel: *.tec [POINT]
- gnuplot: *.tec [POINT]
- MATLAB: *.tec [POINT], *.h5
- Matplotlib: *.tec [POINT]
- Tecplot : *.tec [POINT, BLOCK, FEBRICK]
- ParaView: *.h5, *.xmf, *.vtk
- R *.tec [POINT], *.h5
- VisIt *.h5, *.xmf, *.vtk



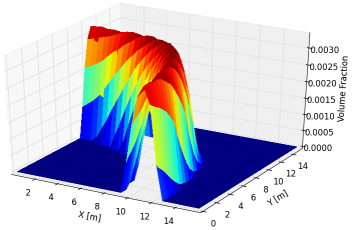
gnuplot



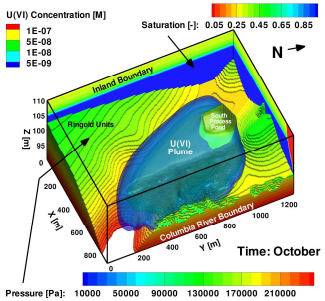
R



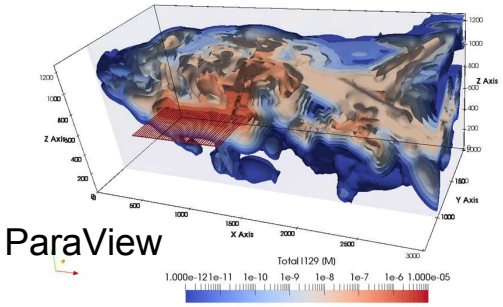
Excel



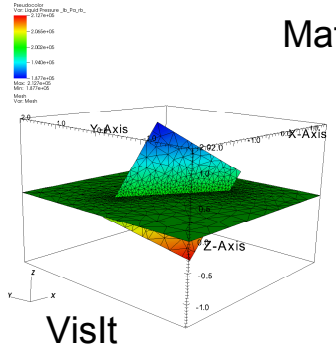
Matplotlib



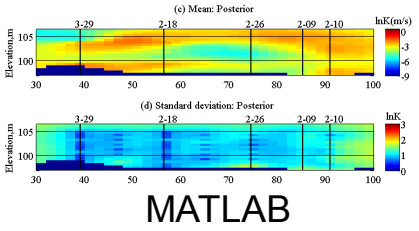
Tecplot



ParaView



VisIt

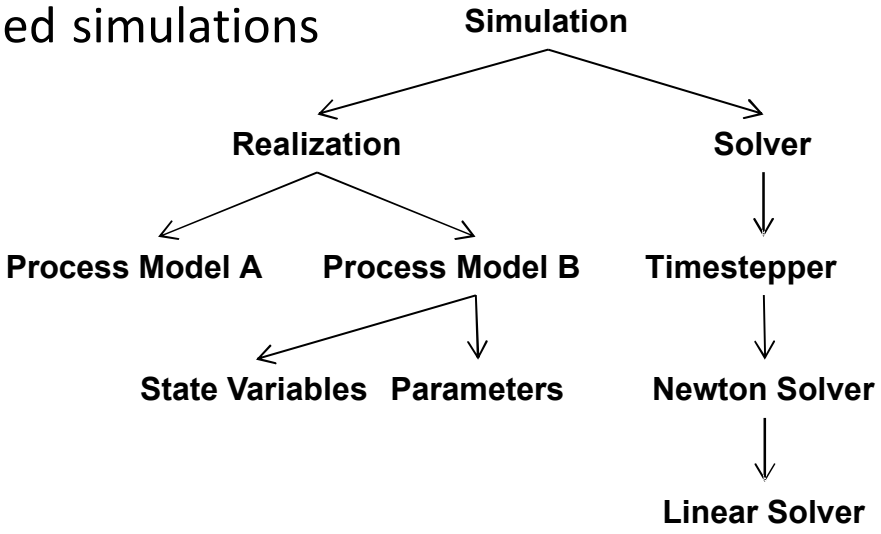


MATLAB

Why Object-Oriented Fortran 2003/2008?

Sandia National Laboratories

- Why Fortran?
  - Experienced domain scientists remain engaged
  - Commonality among all domain scientists
- Why object-oriented?
  - Modular data structures
    - Eases code development and debugging – data locality
    - Nesting of processes and data
  - Tree structure enables self-contained simulations
- Why Fortran 2003/2008?
  - Classes (extendable derived types)
    - Member functions
    - Inheritance
  - Pointers to procedures
    - E.g. swapping equations of state



Object Oriented Fortran

Sandia National Laboratories

Fortran 77

```
subroutine X(a,b,c,d,e,f,…)

...

common/array/a(ncomp,ncell)


do icell = 1, ncell
  do icomp = 1, ncomp
    a(icomp,icell) = …
  enddo
enddo
```

OO Fortran 90

```
subroutine X(realization)

...

grid => realization%patch%grid
reaction => realization%reaction
cells => realization%patch…%cells

do icell = 1, grid%ncell
  do icomp = 1, reaction%ncomp
    cells(icell)%conc(icomp) = …
  enddo
enddo
```

Fortran 2003/2008

Sandia National Laboratories

Fortran 90

```
select case(eos_type)
  case(WATER)
    call EvaluateWater(p,t)
  case(AIR)
    call EvaluateAir(p,t)
  case(CO2)
    call EvaluateCO2(p,t)
  case(CH4)
    call EvaluateCH4(p,t)
end select
```

Fortran 2003/2008

```
type, extends(eos_base) :: eos_CO2
  …
contains
  procedure :: Evaluate => EvaluateCO2
end type eos_CO2

class(eos_CO2) :: eos

call eos%Evaluate(p,t)
```

eos = equation of state

PFLOTRAN Support Infrastructure

Sandia National Laboratories

- 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
      - 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

PFLOTRAN Flow Modes

- Covered in shortcourse
  - RICHARDS: variably-saturated water
  - TH: variably-saturated water-energy
  - GENERAL: multiphase air-water-energy
  - WIPP_FLOW: immiscible air-water
  - MPHASE: supercritical CO2-water-energy
  - FLASH2: supercritical CO2-water-energy (experimental)
  - IMMIS: air-water (experimental)
  - MISCIBLE: X-water-energy (experimental)
  - TOIL_IMS: oil-water-energy (experimental)

Flow Governing Equations

Sandia National Laboratories

General Mode (air-water-energy)

Water Mass

$$\frac{\partial \phi (s_l \rho_l X_w^l + s_g \rho_g X_w^g)}{\partial t} = -\nabla \cdot \left(\rho_l X_w^l \boldsymbol{q}_l + \rho_g X_w^g \boldsymbol{q}_g + \boldsymbol{J}_w^l + \boldsymbol{J}_w^g \right) + q_w$$

Air Mass

$$\frac{\partial \phi (s_l \rho_l X_a^l + s_g \rho_g X_a^g)}{\partial t} = -\nabla \cdot \left(\rho_l X_a^l \boldsymbol{q}_l + \rho_g X_a^g \boldsymbol{q}_g + \boldsymbol{J}_a^l + \boldsymbol{J}_a^g \right) + q_a$$

Energy

$$\frac{\partial \phi \left(s_l \rho_l U_l + s_g \rho_g U_g \right) + (1-\phi) C_p^{\text{rock}} \rho_{\text{rock}} T}{\partial t} = -\nabla \cdot \left(\rho_l H_l \boldsymbol{q}_l + \rho_g H_g \boldsymbol{q}_g - \kappa_{\text{eff}} \nabla T \right) + q_e$$

ϕ = effective porosity [-]
s_l = liquid saturation [-]
s_g = gas saturation [-]
ρ_l = liquid phase density [kmol/m^3]
ρ_g = gas phase density [kmol/m^3]
X_w^l = mole fraction of water in the liquid phase [-]
X_w^g = mole fraction of water in gas phase [-]
X_a^l = mole fraction of air in the liquid phase [-]
X_a^g = mole fraction of air in gas phase [-]
$\boldsymbol{q}_l$ = liquid phase Darcy flux [m/sec]
$\boldsymbol{q}_g$ = gas phase Darcy flux [m/sec]
$\boldsymbol{J}_w^l$ = water diffusive flux in liquid phase [kmol/m^2/sec]
$\boldsymbol{J}_a^l$ = air diffusive flux in liquid phase [kmol/m^2/sec]
$\boldsymbol{J}_w^g$ = water vapor diffusive flux in gas phase [kmol/m^2/sec]
$\boldsymbol{J}_a^g$ = air diffusive flux in gas phase [kmol/m^2/sec]
q_w = water source/sink [kmol/sec]
q_a = air source/sink [kmol/sec]
q_e = energy source/sink [MJ/sec]
U_l = liquid phase internal energy [MJ/kmol]
U_g = gas phase internal energy [MJ/kmol]
H_l = liquid phase enthalpy [MJ/kmol]
H_g = gas phase enthalpy [MJ/kmol]
C_p^{rock} = rock heat capacity [MJ/kg rock-K]
ρ_{rock} = rock particle density [kg/m^3 rock]
T = temperature [C]
κ_{eff} = effective thermal conductivity [W/K-m]

Flow Governing Equations

Darcy Flux

$$\boldsymbol{q}_\alpha = -\frac{k k_\alpha}{\mu_\alpha}\nabla(p_\alpha - \gamma_\alpha g z), \quad (\alpha = l,\ g)$$

Air Diffusion in Liquid Phase

$$\boldsymbol{J}_a^l = -\tau\phi s_l D_l \rho_l \boldsymbol{\nabla} X_a^l$$

Air Diffusion in Gas Phase

$$\boldsymbol{J}_a^g = -\tau\phi s_g D_g^0 \left(\frac{T}{T_K}\right)^\theta \frac{p_0}{p_g}\rho_g \nabla X_a^g$$

$\boldsymbol{q}_\alpha$ = Darcy flux for phase α [m/s]
k = intrinsic permeability [m^2]
k_α = relative permeability for phase α [-]
μ_α = viscosity for phase α [Pa-s]
p_α = pressure for phase α [Pa]
γ_α = density for phase α [kg/m^3]
g = gravity [m/s^2]
z = elevation [m]
$\boldsymbol{J}_a^l$ = diffusive flux of air in liquid phase [kmol/m^2-s]
τ = tortuosity [-]
D_l = aqueous diffusivity [m^2/s]
$\boldsymbol{J}_a^g$ = diffusive flux of air in gas phase [kmol/m^2-s]
D_g^0 = gas diffusivity [m^2/s]
p_0 = reference pressure [Pa]

Flow Primary Variables

General Mode

Thermodynamic State of Fluid	Primary Variables X_1	X_2	X_3
Two-Phase	$\boldsymbol{p_g}$	$\boldsymbol{s_g}$	$\boldsymbol{T}$
	p_g	s_g	p_a
Liquid	$\boldsymbol{p_l}$	$\boldsymbol{X_a^l}$	$\boldsymbol{T}$
Gas	$\boldsymbol{p_g}$	$\boldsymbol{p_a}$	$\boldsymbol{T}$

Flow Governing Equations

Sandia National Laboratories

TH Mode (water-energy)

Water Mass

$$\frac{\partial \phi(s_l \rho_l)}{\partial t} = -\nabla \cdot (\rho_l \boldsymbol{q}_l) + q_w$$

Energy

$$\frac{\partial \phi \left(s_l \rho_l U_l\right) + (1-\phi) C_p^{\text{rock}} \rho_{\text{rock}} T}{\partial t} = -\nabla \cdot (\rho_l H_l \boldsymbol{q}_l - \kappa_{\text{eff}} \nabla T) + q_e$$

Flow Governing Equations

Sandia National Laboratories

RICHARDS Mode (water)

Water Mass

$$\frac{\partial \phi(s_l \rho_l)}{\partial t} = -\nabla \cdot (\rho_l \boldsymbol{q}_l) + q_w$$

Constitutive Relations

- Characteristic curves
  - Capillary pressure / saturation functions
    - E.g. Brooks Corey, van Genutchen
  - Relative permeability functions
    - E.g. Burdine, Mualem
  - One may add custom characteristic curves by extending the base classes

Saturation Function Classes

```
type :: sat_func_base_type
   type(polynomial_type), pointer :: sat_poly
   type(polynomial_type), pointer :: pres_poly
   PetscReal :: Sr
   PetscReal :: pcmax
   PetscBool :: analytical_derivative_available
 contains
   procedure, public :: Init => SFBaseInit
   procedure, public :: Verify => SFBaseVerify
   procedure, public :: Test => SFBaseTest
   procedure, public :: SetupPolynomials => SFBaseSetupPolynomials
   procedure, public :: CapillaryPressure => SFBaseCapillaryPressure
   procedure, public :: Saturation => SFBaseSaturation
 end type sat_func_base_type

 type, public, extends(sat_func_base_type) :: sat_func_VG_type
   PetscReal :: alpha
   PetscReal :: m
 contains
   procedure, public :: Init => SF_VG_Init
   procedure, public :: Verify => SF_VG_Verify
   procedure, public :: CapillaryPressure => SF_VG_CapillaryPressure
   procedure, public :: Saturation => SF_VG_Saturation
 end type sat_func_VG_type
```

Constitutive Relations (cont.)

- Equations of state
  - Water
    - Density (Constant, Exponential, Planar (lookup table), **IFC67**, Batzle and Wang,, etc.)
    - Enthalpy (Constant, Planar, **IFC67**, etc.)
    - Viscosity (Constant, **Default**, Batzle and Wang, Grabowski)
  - Steam
    - Density (Constant, Planar, **IFC67**)
    - Enthalpy (Constant, Planar, **IFC67**)
  - Gas
    - Density (Constant, **Ideal**, RKS)
    - Enthalpy (Constant, **Ideal**)
    - Viscosity (Constant, **TOUGH2** mixture)
  - Custom EOSs can be added through the use of Fortran procedure pointers

Constitutive Relations (cont.)

- Solubility of gas component in liquid phase
  - Function of gas pressure through the Henry's law constant
- Water saturation pressure
  - IFC67

Transport

- Multicomponent
- Mobile/immobile primary species
- Advection
  - First order upwinding
- Hydrodynamic dispersion
  - Temperature-dependent molecular diffusion
  - Species-dependent molecular diffusion (charge balance issues)
  - Material-specific mechanical dispersion

Biogeochemical Reaction

- Aqueous speciation
  - Ion activity models
    - Debye-Huckel B-dot
- Mineral precipitation-dissolution
  - Kinetic
    - Transition state theory
    - Temperature dependent
      - Arrhenius-based
    - Prefactors
    - Attenuation
    - Rate limiters
    - Thresholds
  - Equilibrium – initial/boundary conditions only
- General ($A + B \leftrightarrow C$)
  - N^{th} order kinetics
  - Reversible
- Microbiological
  - Michaelis–Menten kinetics
  - Biomass
  - Inhibition
- Radioactive decay with daughter products
  - Limited to aqueous and sorbed phases

Biogeochemical Reaction (cont.)

- Sorption
  - Isotherm-based
    - Linear
    - Langmuir
    - Freundlich
  - Ion exchange
    - Equilibrium
  - Surface complexation
    - Equilibrium
    - Kinetic
    - Multirate kinetic
- Reaction Sandbox
  - Framework for implementing custom kinetic (rate-limited) reactions

Reactive Transport Governing Equations

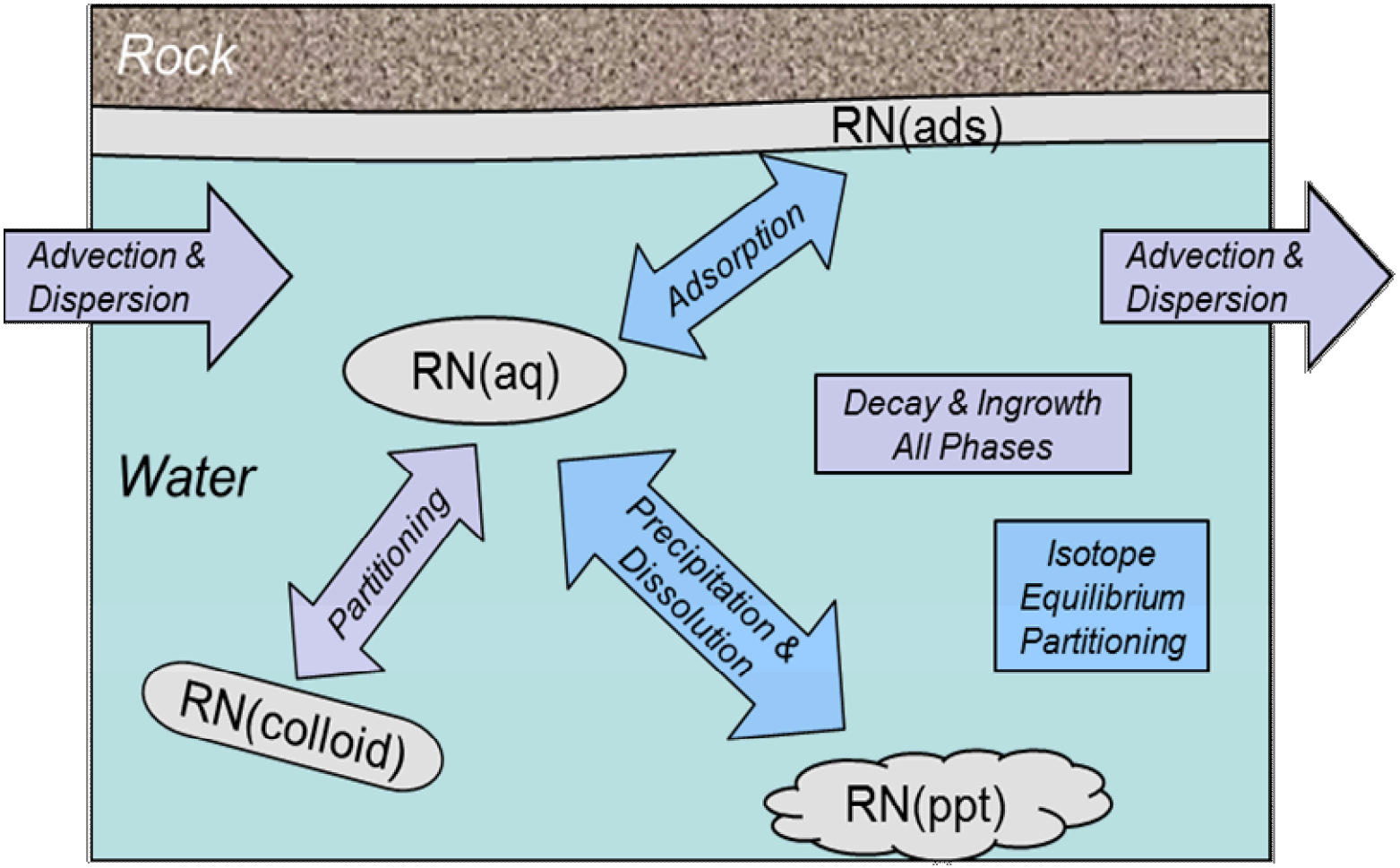
$$\frac{\partial}{\partial t}\left(\phi s \Psi_j + \Psi_{S,j}\right) + \nabla \cdot \left(\boldsymbol{q} - \phi s \boldsymbol{D} \boldsymbol{\nabla}\right) \Psi_j = Q_j - \sum_r \nu_{jr} I_r$$

$$\Psi_j = C_j + \sum_{i=1}^{N_X} \nu_{ji} \frac{K_i}{\gamma_i} \prod_{j'=1}^{N_c} \left(\gamma_{j'} C_{j'}\right)^{\nu_{j'i}}$$

ϕ = porosity [-]
s = liquid saturation [-]
Ψ_j = total component concentration of species j [mol/L]
$\Psi_{S,j}$ = total sorbed concentration of species j [mol/m^3]
$\boldsymbol{q}$ = liquid Darcy velocity [m/s]
$\boldsymbol{D}$ = hydrodynamic dispersion coefficient [m^2/s]
Q_j = source/sink for species j [mol/s]
ν_{jr} = stoichiometry of species j in kinetic reaction r [-]

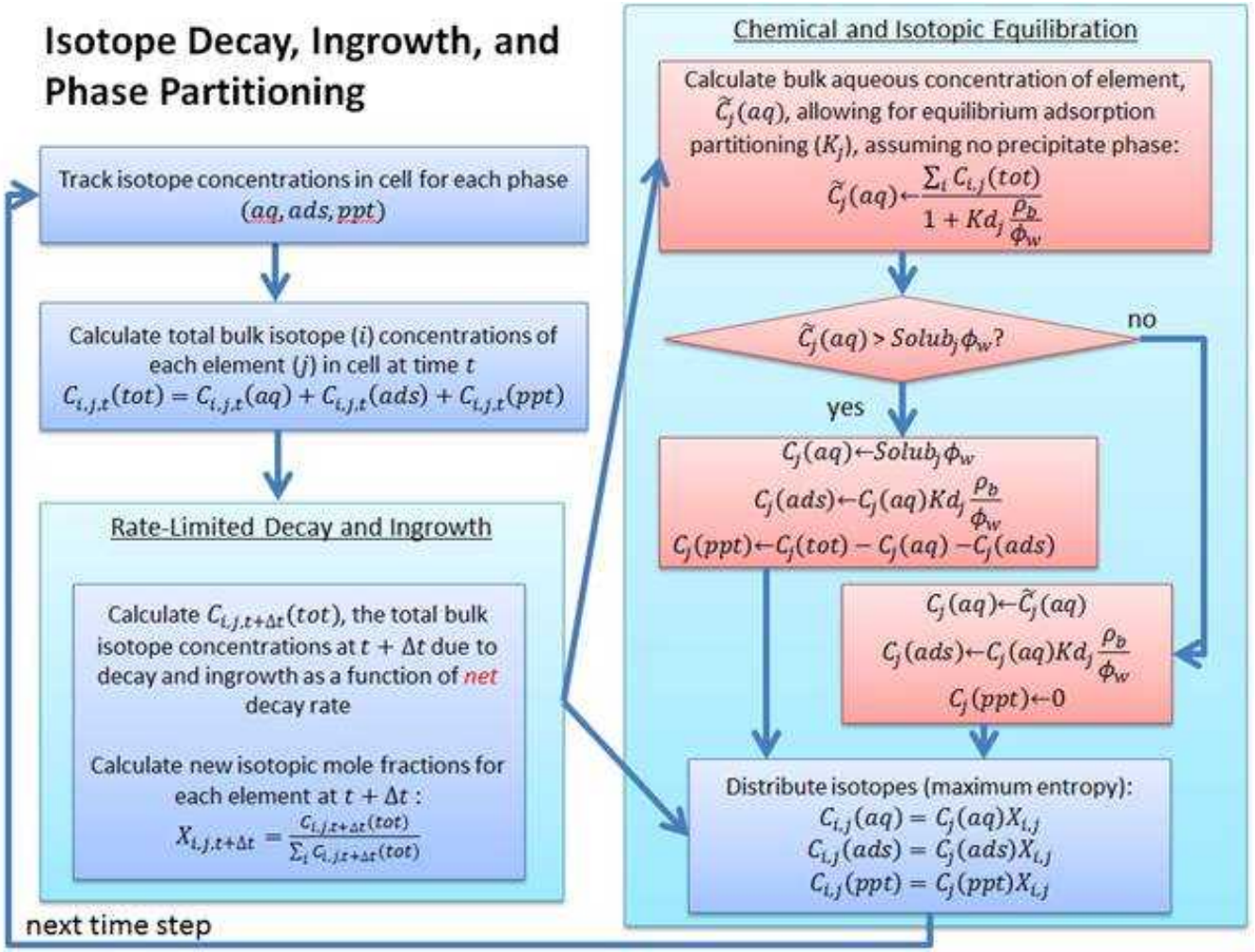
I_r = kinetic rate for reaction r [mol/s]
C_j = free ion concentration for species j [mol/kg water]
N_X = # aqueous complexes [-]
ν_{ji} = stoichiometry of species j in complex i [-]
K_i = equilibrium constant for complex i
γ_i = activity coefficient for complex i [-]
N_c = # primary aqueous species [-]

Isotope Decay, Ingrowth and Partitioning



Paul Mariner
Sandia National Laboratories

Isotope Decay, Ingrowth and Partitioning



Paul Mariner
Sandia National Laboratories

Reaction Sandbox

Sandia National Laboratories

Reaction Sandbox Class (to be modified by the user)

```
type, public, extends(reaction_sandbox_base_type) :: &
                                reaction_sandbox_xxx_type
  ! Add class parameters here
  class(reaction_sandbox_base_type), pointer :: next
contains
  ! Map procedures here
  procedure, public :: ReadInput => XXXRead
  procedure, public :: Setup => XXXSetup
  procedure, public :: Evaluate => XXXReact
  procedure, public :: Destroy => XXXDestroy
end type reaction_sandbox_base_type
```

Reaction Sandbox Linked List (not modified by the user)

```
cur_reaction => rxn_sandbox_list
do
  if (.not.associated(cur_reaction)) exit
  call cur_reaction%Evaluate(Residual,Jacobian, &
                              state_variables)
  cur_reaction => cur_reaction%next
enddo
```

Simple Reaction Sandbox

Sandia National Laboratories

Available Species:

Aqueous : `Aaq, Baq, Caq, Daq, Eaq, Faq`

Immobile: `Xim, Yim`

Example Rate Expressions:

Units of Rate Constant k

```
Rate = k * L_water                                  ! [mol/L-sec]
Rate = k * Aaq * L_water                            ! [1/sec]
Rate = k * Aaq * Baq * L_water                      ! [L/mol-sec]
Rate = k * Aaq/(K_Aaq + Aaq) * L_water              ! [mol/L-sec]
Rate = k * Xim * L_water * &                        ! [mol-m^3 bulk /
        Aaq/(K_Aaq + Aaq) * &                       !   L-mol_biomass-sec]
        Baq/(K_Baq + Baq)
Rate = (k * Aaq – kr * Caq) * L_water               ! [1/sec]
```

0th Order

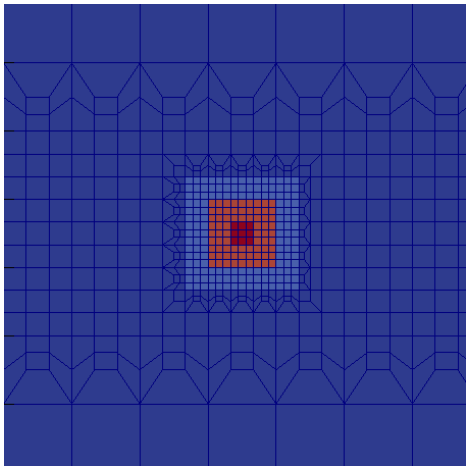
1st Order

2nd Order

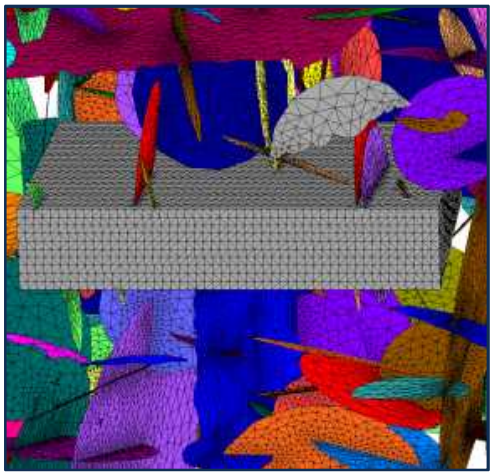
Assumed Units: Aqueous species [mol/L water], Immobile species [mol/m^3 bulk volume], Rate constant k [reaction-dependent], Rate [mol/sec], K_Xxx [mol/L water], L_water [L water]

Gridding

- Structured
  - Uniform or variable grid spacing
- Unstructured Implicit
  - Finite element format (elements and vertices)
- Unstructured Explicit
  - Polyhedral format (define cells and connectivity)



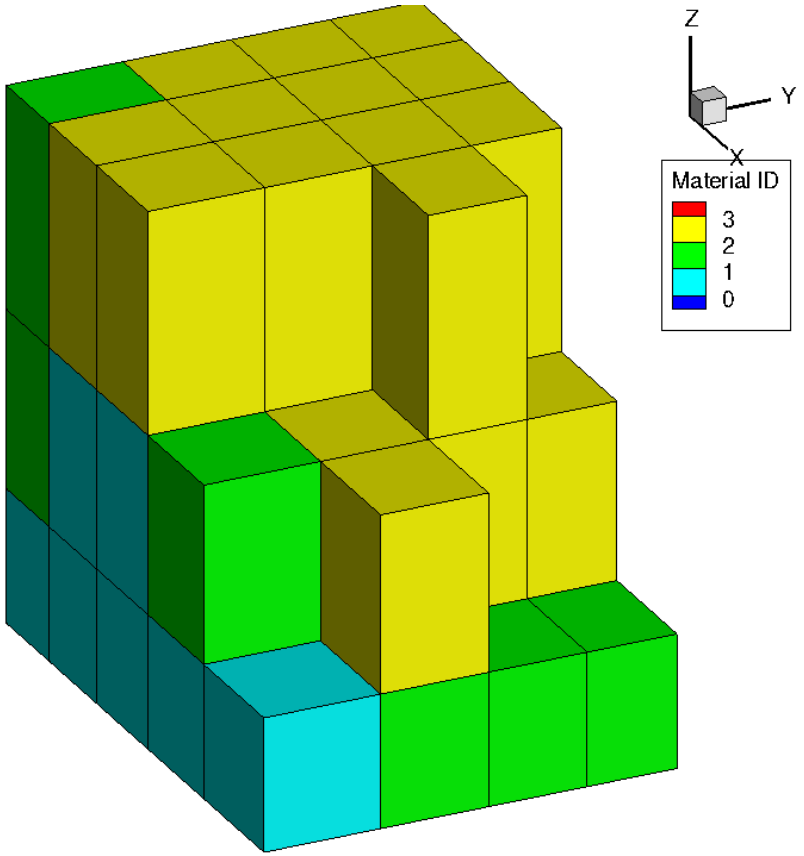
Emily Stein
Sandia National Laboratories



Satish Karra
Los Alamos National Laboratory

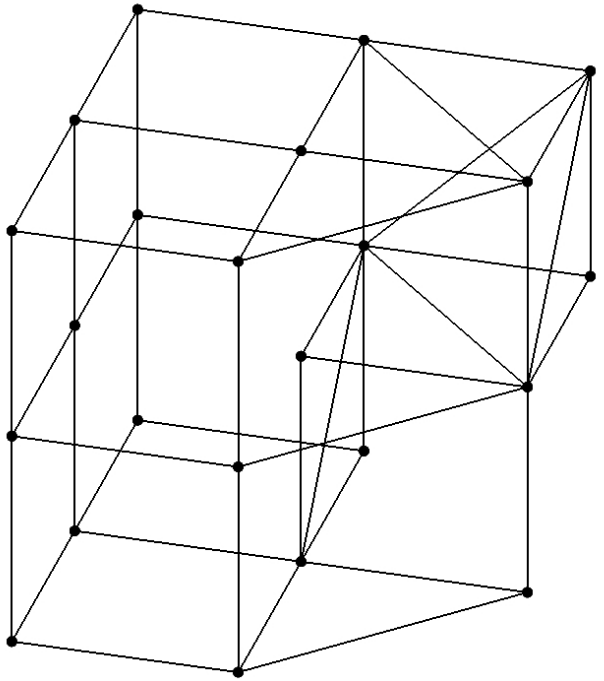
Structured Grid

```
GRID
  TYPE structured
  ORIGIN 0.d0 0.d0 0.d0
  NXYZ 5 4 3
  DXYZ
    10. 11. 12. 13. 14.
    13. 12. 11. 10.
    15. 20. 25.
  /
END
```



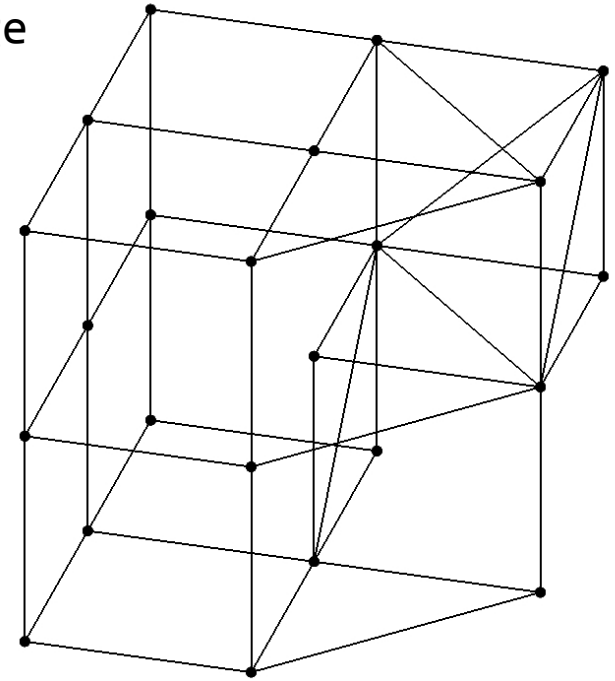
Unstructured Implicit Grid

```
GRID
  TYPE unstructured ./mixed.h5
END


15 24
P 4 5 6 2 1
T 4 3 5 1
H 19 9 5 12 17 7 6 16
T 5 13 14 15
P 13 5 12 22 15
H 20 10 9 19 18 8 7 17
...
5.000000e+00 5.000000e+00 5.000000e+00
5.000000e+00 2.500000e+00 5.000000e+00
5.000000e+00 5.000000e+00 2.500000e+00
5.000000e+00 2.500000e+00 2.500000e+00
...
0.000000e+00 2.500000e+00 0.000000e+00
```



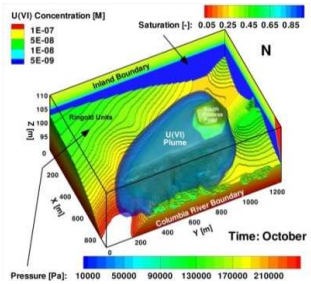
Unstructured Explicit Grid

```
GRID
  TYPE unstructured_explicit ./mixed.uge
END

CELLS 15
1 4.0625 4.0625 4.0625 5.20833
2 4.375 4.375 3.125 2.60417
3 3.3333 3.3333 3.75 7.8125
4 3.3333 1.6667 3.75 7.8125
5 3.3333 1.6667 1.25 7.8125
...
15 1.25 3.75 0.3125 2.60417
CONNECTIONS 24
1 2 4.16667 4.16667 3.3333 6.25
1 3 3.75 3.75 3.75 8.8388
3 4 3.75 2.5 3.75 6.25
...
14 15 0.41667 3.75 0.41667 2.2097
```

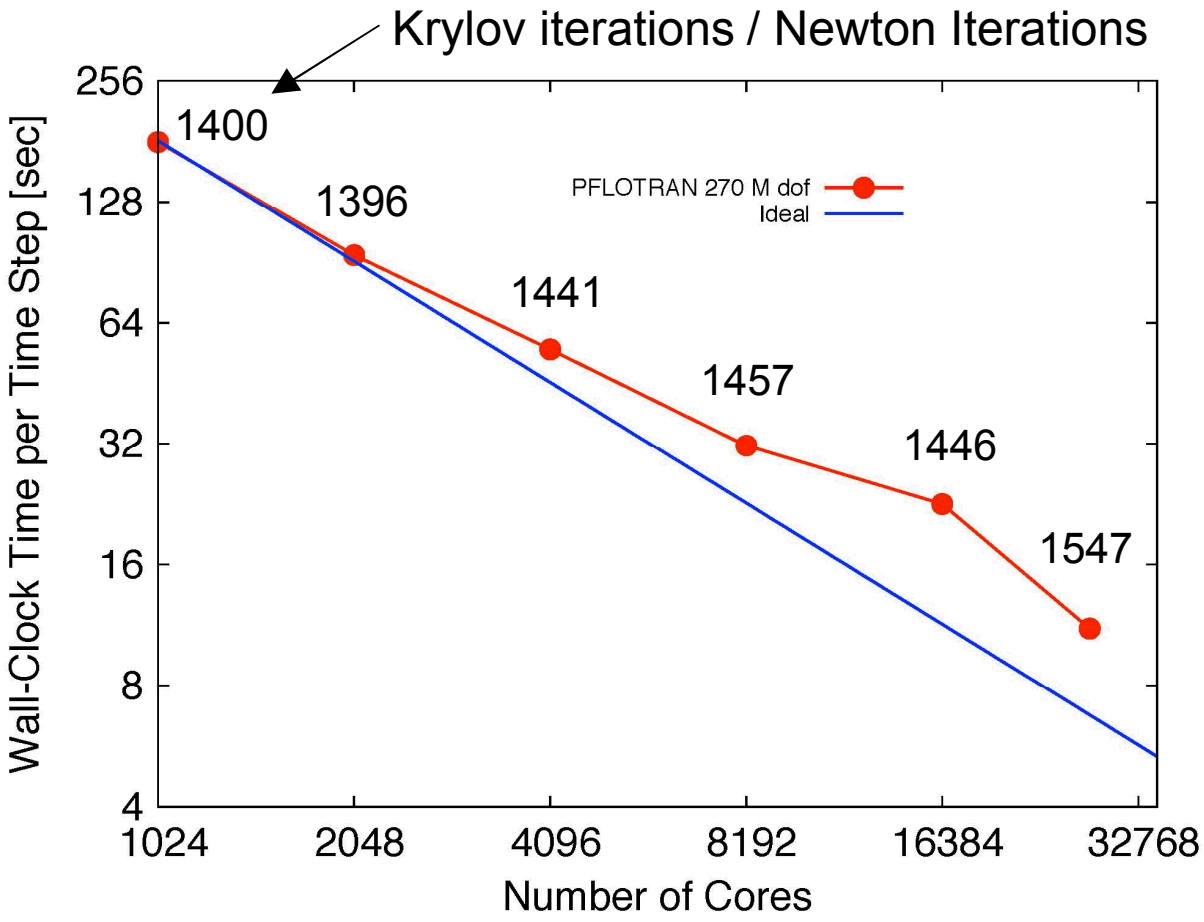


PFLOTRAN Parallel Performance

Hanford 300 Area



Hammond and Lichtner, WRR, 2010



Overall 60% efficient

Science

(1-2% of overall time)

- 75% efficient (wall clock)

Solvers

(96% of overall time)

- 90% efficient (iteration count)
- 64% efficient (wall clock)

Sandia National Laboratories

ANDRA QUESTIONS

ANDRA Questions

- Multiphase Flow
  - Solvers employed (in particular, for variably saturated transport)?
    - Nonlinear solver
      - Newton's method
        - » Analytical and/or numerical derivatives
    - Linear solver
      - Direct
      - Iterative Krylov
      - Constrained Pressure Residual (CPR) being investigated for oil flow modes
  - How is multiphase state transition handled?
    - Swapping of primary dependent variables when thresholds are exceeded
    - Thresholds
      - Liquid $\rightarrow$ 2 Phase vapor pressure < saturation pressure
      - Gas $\rightarrow$ 2 Phase vapor pressure > saturation pressure
      - 2 Phase $\rightarrow$ Liquid gas saturation < 0
      - 2 Phase $\rightarrow$ Gas gas saturation > 1

ANDRA Questions

- Multiphase Flow
  - How are molecular diffusion terms defined? By phase, species, material, zone?
    - Multiphase flow
      - Molecular diffusion: by phase
    - Reactive transport
      - Molecular diffusion: by species and phase
      - Mechanical dispersion: by material
  - How is dissolution of gas component in liquid phase handled? Henry's law?
    - Henry's law
  - How are thermal dynamic properties of water handled?
    - Through equations of state for each phase

ANDRA Questions

- Reactive Transport
  - Kinetic precipitation-dissolution?
    - Yes, through kinetic mineral precipitation-dissolution
  - Ion exchange and surface complexation?
    - Yes
  - Is it possible to simulate reactive transport in variably saturated media?
    - Yes, saturation state has no impact on biogeochemical reaction
  - What phenomena are currently coupled to chemistry?
    - Multiphase flow
    - Multicomponent transport
    - Density dependent flow (e.g. salinity)
    - Feedbacks to flow/transport through porosity and tortuosity

ANDRA Questions

- Heterogeneous material properties (how assigned)?

How are initial and boundary conditions and material properties assigned?

Sandia National Laboratories

- Create definition blocks
  - Material properties
  - Flow conditions
  - Transport conditions
  - Regions
- Create couplers
  - Initial condition
  - Boundary condition
  - Source sink
  - Strata

How are material properties assigned?

- Define a material property blocks

```
MATERIAL_PROPERTY sand
  ID 1
  POROSITY DATASET Porosity
  TORTUOSITY 1.d0
  CHARACTERISTIC_CURVES sf1
  PERMEABILITY
    PERM_X 1.d-12
    PERM_Y 1.d-12
    PERM_Z 1.d-13
  /
END

MATERIAL_PROPERTY clay
  ID 2
  POROSITY 0.55d0
  TORTUOSITY DATASET Tortuosity
  CHARACTERISTIC_CURVES sf2
  PERMEABILITY
    DATASET Permeability
  /
END
```

How are material properties assigned?

- Define region blocks

```
REGION all
  COORDINATES
    0.d0 0.d0 0.d0
    60.d0 46.d0 60.d0
  /
END

REGION east
  FACE EAST
  COORDINATES
    60.d0 0.d0 0.d0
    60.d0 46.d0 60.d0
  /
END

REGION well
  BLOCK 4 4 2 3 3 3
END

REGION observation_point
  COORDINATE 35.d0 22.d0 15.d0
END
```

How are material properties assigned?

- Couple them through strata blocks

```
STRATA
  REGION all
  MATERIAL clay
END

STRATA
  REGION well
  MATERIAL sand
END

STRATA
  MATERIAL materials_ids.h5
END
```

Material IDs defined on a cell by cell basis

How are initial and boundary conditions assigned?

- Define flow condition blocks

```
FLOW_CONDITION east
  TYPE
    PRESSURE hydrostatic
  /
  INTERPOLATION LINEAR
  DATUM LIST
    TIME_UNITS d
    0.d0 0.d0 0.d0 34.d0
    10.d0 0.d0 0.d0 39.d0
    50.d0 0.d0 0.d0 33.d0
    100.d0 0.d0 0.d0 34.d0
  /
  PRESSURE 101325 ! Pa
END

FLOW_CONDITION injection_well
  TYPE
    RATE scaled_volumetric_rate neighbor_perm
  /
  RATE 1 m^3/hr
END
```

How are initial and boundary conditions assigned?

- Define transport condition blocks

```
TRANSPORT_CONDITION columbia_river
  TYPE dirichlet_zero_gradient
  CONSTRAINT_LIST
    0.d0 river_water
  /
END

TRANSPORT_CONDITION initial_groundwater
  TYPE dirichlet_zero_gradient
  CONSTRAINT_LIST
    0.d0 initial_groundwater
  /
END

TRANSPORT_CONDITION tracer_pulse
  TYPE dirichlet
  TIME_UNITS d
  CONSTRAINT_LIST
    0.d0 well_tracer
    10.d0 groundwater
  /
END
```

How are initial and boundary conditions assigned?

- Define constraint blocks for use within transport conditions

```
CONSTRAINT initial_groundwater
  CONCENTRATIONS
    H+        1.d-7            M Calcite
    Ca++      1.20644e-3       T
    Cu++      1.e-8            T
    Mg++      5.09772e-4       T
    UO2++     2.4830E-11       T DATASET initial_U
    K+        1.54789e-4       T
    Na+       1.03498e-3       Z
    HCO3-     2.57305e-3       T
    Cl-       6.97741e-4       T
    F-        2.09491e-5       T
    HPO4--    1.e-8            M Fluorapatite
    NO3-      4.69979e-4       T
    SO4--     6.37961e-4       T
  /
  MINERALS
    Calcite   DATASET initial_Calcite_vol_frac DATASET initial_Calcite_area
    Metatorbernite 0.    1. cm^2/cm^3
  /
END
```

How are initial and boundary conditions assigned?

- Couple them through boundary and initial condition blocks

```
INITIAL_CONDITION
  FLOW_CONDITION initial
  TRANSPORT_CONDITION initial_groundwater
  REGION all
END

BOUNDARY_CONDITION east
  FLOW_CONDITION east
  TRANSPORT_CONDITION columbia_river
  REGION east
END

SOURCE_SINK well
  FLOW_CONDITION injection_well
  TRANSPORT_CONDITION tracer_pulse
  REGION well
END
```

ANDRA Questions

- Heterogeneous material properties (how assigned)?

```
CONSTRAINT initial_groundwater
  CONCENTRATIONS
    H+        1.d-7            M Calcite
    Ca++      1.20644e-3       T
    Cu++      1.e-8            T
    Mg++      5.09772e-4       T
    UO2++     2.4830E-11       T DATASET initial_U
    K+        1.54789e-4       T
    Na+       1.03498e-3       Z
    HCO3-     2.57305e-3       T
    Cl-       6.97741e-4       T
    F-        2.09491e-5       T
    HPO4--    1.e-8            M Fluorapatite
    NO3-      4.69979e-4       T
    SO4--     6.37961e-4       T
/
  MINERALS
    Calcite   DATASET initial_Calcite_vol_frac DATASET initial_Calcite_area
    Metatorbernite 0.    1. cm^2/cm^3
  /
END
```

ANDRA Questions

- Decay chains (with and without branching)?
  - For aqueous and sorbed phases, use standard reactive transport
  - For aqueous, sorbed and mineral phases, use Isotope Decay, Ingrowth and Partitioning process model
- Sorption (isotherms supported)
  - Linear, Langmuir and Freundlich
- Precipitation-dissolution (handing of radioactive decay in solid phase)
  - Isotope Decay, Ingrowth and Partitioning process model

ANDRA Questions

- Boundary and initial conditions (variable in time and space)?
  - Boundary conditions:
    - Variable in time: yes
    - Variable in space: yes
      - By region
      - Interpolated
  - Initial conditions: variable in space
    - Variable in space: yes
      - By region
      - Cell by cell
      - Interpolated
- Gridding (irregular Cartesian allowed?)
  - Yes
    - Structured with variable spacing
    - Unstructured explicit

ANDRA Questions

- General
  - Is it possible to calculate a flux across an internal surface?
    - Currently restricted to Cartesian grid
    - Plans to implement unstructured
  - Is full tensor hydraulic conductivity supported?
    - No, nor full tensor dispersion either
  - Are some of these points scheduled for a future version of PFLOTRAN?
    - Fluxes across internal surfaces for unstructured grids (fall 2017)

ANDRA Questions

- Roadmap for future development
  - Fluxes across internal surfaces for unstructured grids (fall 2017)
  - Colloids - kinetic rate limited (2018)
  - Input file interface for species dependent diffusion coefficients (2018)

Questions?

Sandia National Laboratories