HydroGeoSphere
The Basics

Installation Package

Model Setup and Execution

Hands on Examples

Installation Package

- **Default installation directory C:\Program_Files\HydroGeoSphere**
- **Includes: Documentation, Illustration and Verification examples, and all required executables.**

Name	Date modified	Type	Size
docs	11/03/2014 9:54 AM	File folder	
Illustration	11/03/2014 9:54 AM	File folder	
verification	11/03/2014 2:16 PM	File folder	
grok.exe	11/03/2014 9:16 AM	Application	4,479 KB
hsbatch.exe	11/03/2014 9:19 AM	Application	727 KB
hsplot.exe	11/03/2014 9:19 AM	Application	1,141 KB
phgs.exe	11/03/2014 9:19 AM	Application	3,412 KB
rlmhostid.exe	15/01/2013 6:41 PM	Application	1,121 KB
Uninstall.exe	11/03/2014 9:54 AM	Application	152 KB
libifcoremd.dll	22/07/2011 12:15 ...	Application extens...	1,284 KB
libiomp5md.dll	22/07/2011 12:15 ...	Application extens...	824 KB
libmmd.dll	22/07/2011 12:15 ...	Application extens...	3,178 KB
hgs.lic	03/05/2013 1:03 PM	LIC File	1 KB
hostid.txt	11/03/2014 9:54 AM	TXT File	1 KB
rlmhost.bat	11/03/2014 9:54 AM	Windows Batch File	1 KB

The Manuals

- Theory Manual
  - A review of the hydrologic theory/numerical implementation powering HGS
- Reference Manual
  - A handy reference guide to all Grok commands/features/instructions
  - Chapter 1 – Installation guide, review of executables
  - Chapter 2 – Review of commands (i.e. how to write the *.grok file)
    - Mesh Generation
    - Commands for interacting with the mesh
    - Initial and boundary condition assignment
    - Material property assignment
    - Output control
    - Etc….
- Verification Manual
  - A summary of verification problems for testing and learning

What Do the Executables Do?

- **HydroGeoSphere consists of 3 Key Executables**
  1. **grok.exe – Compiles *.grok file which contains the model definition and setup information. Prepares all information for HGS**
  2. **phgs.exe – Serial or parallel numerical simulation**
  3. **hsplot.exe – Processes model output into a format readable by 3D visualization products (e.g., Tecplot)**

Basic HydroGeoSphere Workflow

- **To use HGS you must follow this workflow:**
  1) Configure/write the **.grok* file
  2) Run the *grok.exe* executable
  3) Run the *phgs.exe* executable
  4) Run the *hsplot.exe* (or *hs2vtu.exe*) executable
  5) Review and visualize results

1) Configure/write the *.grok file

- **.grok* files contains instructions used to build the model files.
- All input commands are placed into the **.grok* file.
- Commands are described in the HGS reference manual, which you can find in your doc directory.
- > 30 sample **.grok* files (with associated data) are included on installation
  - C:\Program_Files\HydroGeoSphere\verification

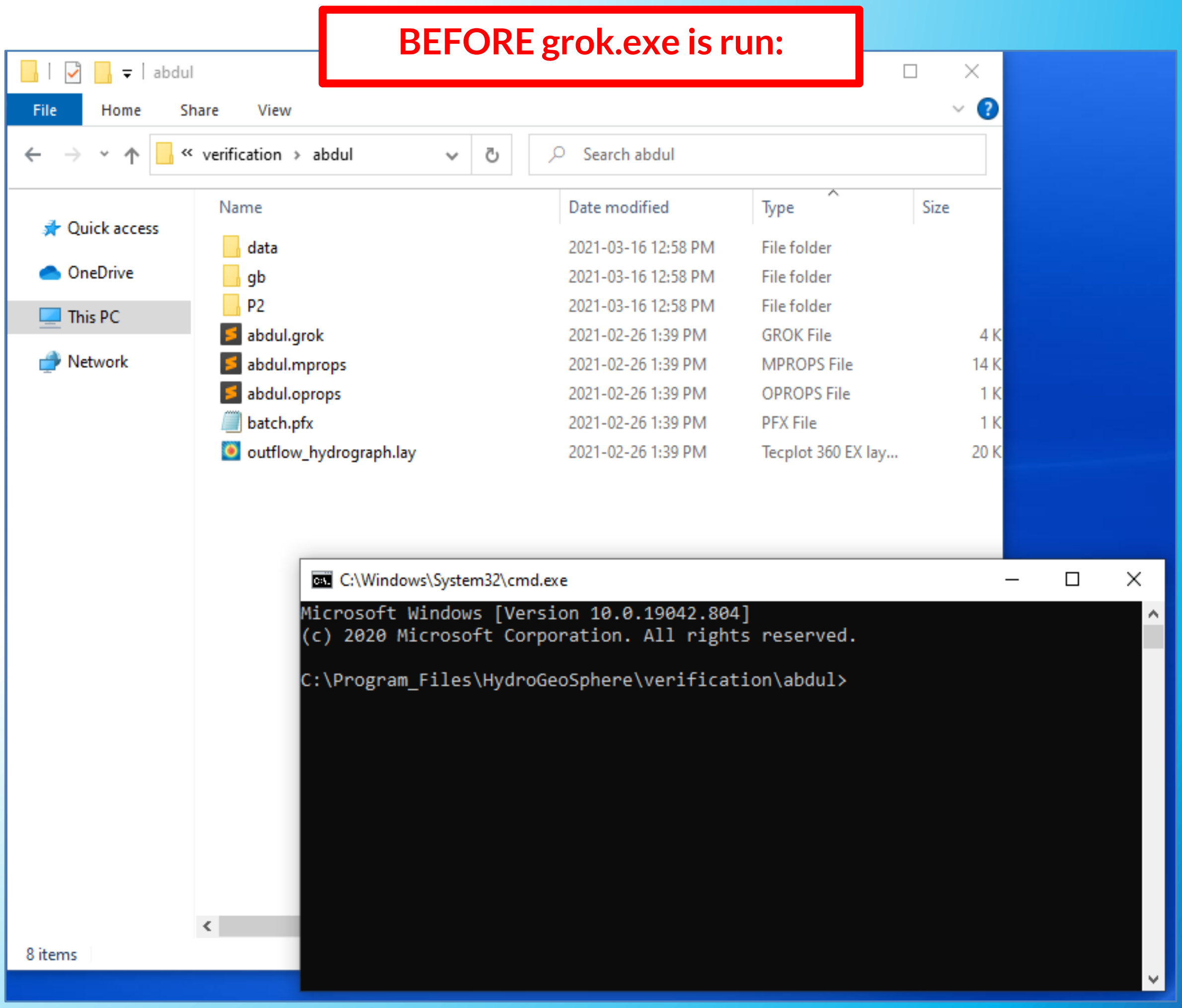
Sample **.grok* file :

abdul.grok

```
!------------------------- Problem description
Overland flow example 2
abdul, Monday, July 25, 2005 at 09:42
end title

!------------------------- Grid generation
read gb 2d grid
./gb/grid

generate layers interactive
    zone by layer

    new layer
        layer name
        Base layer

        uniform sublayering
        5

        elevation from gb file
        ./gb/grid.nprop.Surface elevation - 1.5 m
    end

    new layer
        layer name
        Middle layer

        uniform sublayering
        5

        elevation from gb file
        ./gb/grid.nprop.Surface elevation - 0.5 m
```

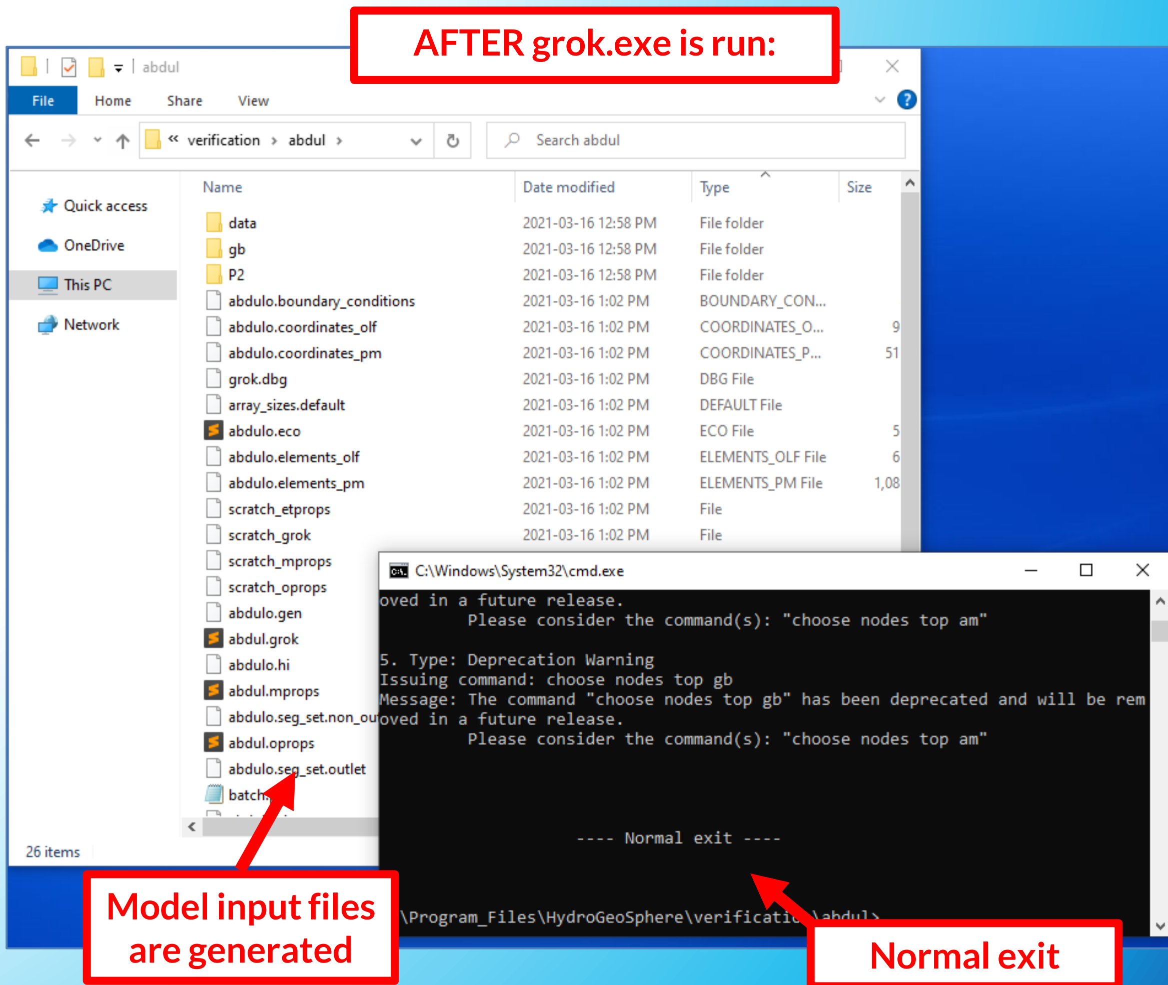
2) Run the grok.exe executable (BEFORE)

- *grok.exe* is a program which converts the **.grok* file into the input files required to run HGS
- Use the command prompt (*cmd.exe*) to run *grok.exe*
  - 1) Go to project folder
  - 2) type '*cmd*' in address bar and press enter (↵)
  - 3) type '*grok.exe*', press enter (↵)
  - 4) *grok* progress is displayed
  - 5) *grok* creates input files for *phgs.exe*
  - 6) "---- normal exit----"



2) Run the grok.exe executable (AFTER)

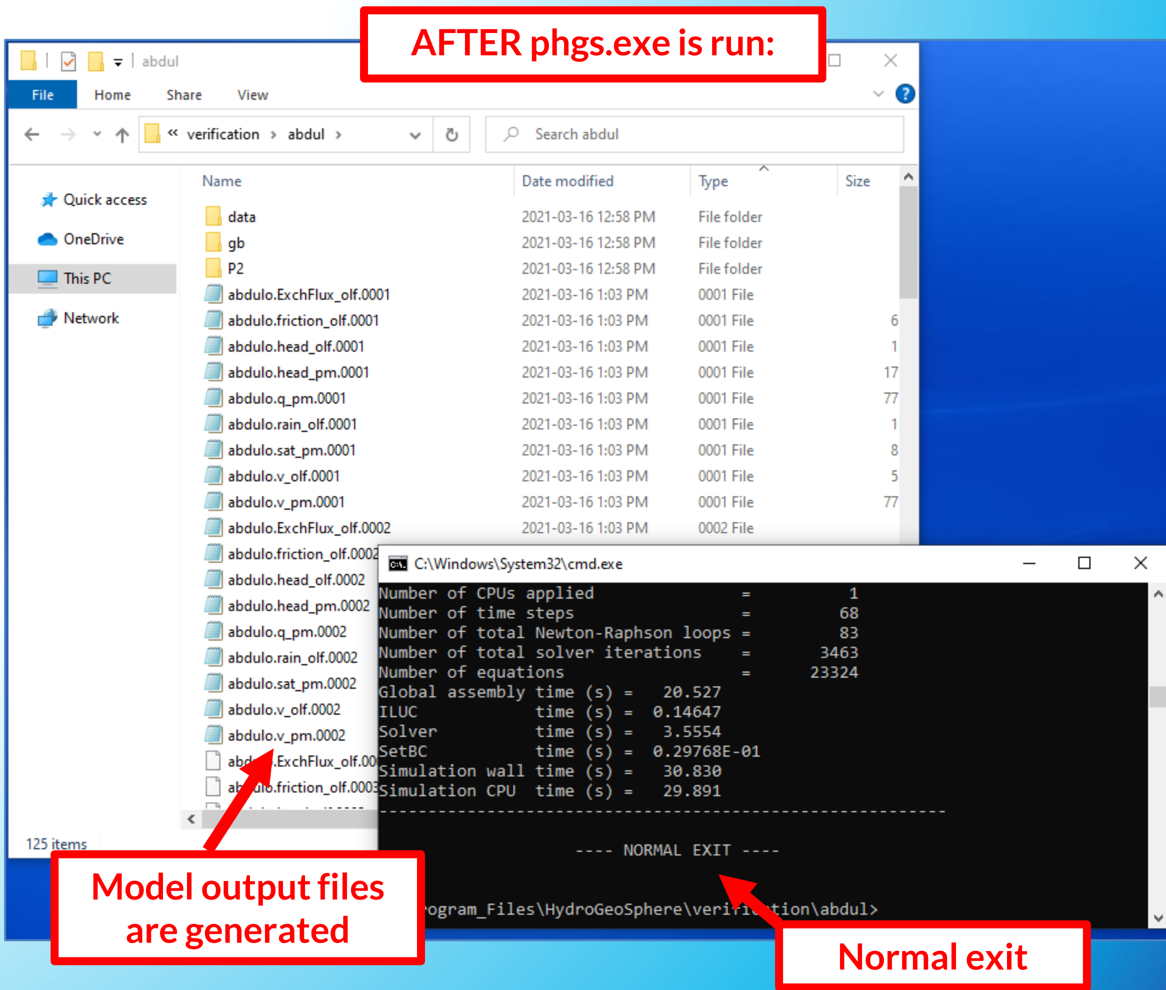
- *grok.exe* is a program which converts the **.grok* file into the input files required to run HGS
- Use the command prompt (*cmd.exe*) to run *grok.exe*
  - 1) Go to project folder
  - 2) type '*cmd*' in address bar and press enter (↵)
  - 3) type '*grok.exe*', press enter (↵)
  - 4) *grok* progress is displayed
  - 5) *grok* creates input files for *phgs.exe*
  - 6) "---- normal exit----"



aquanty

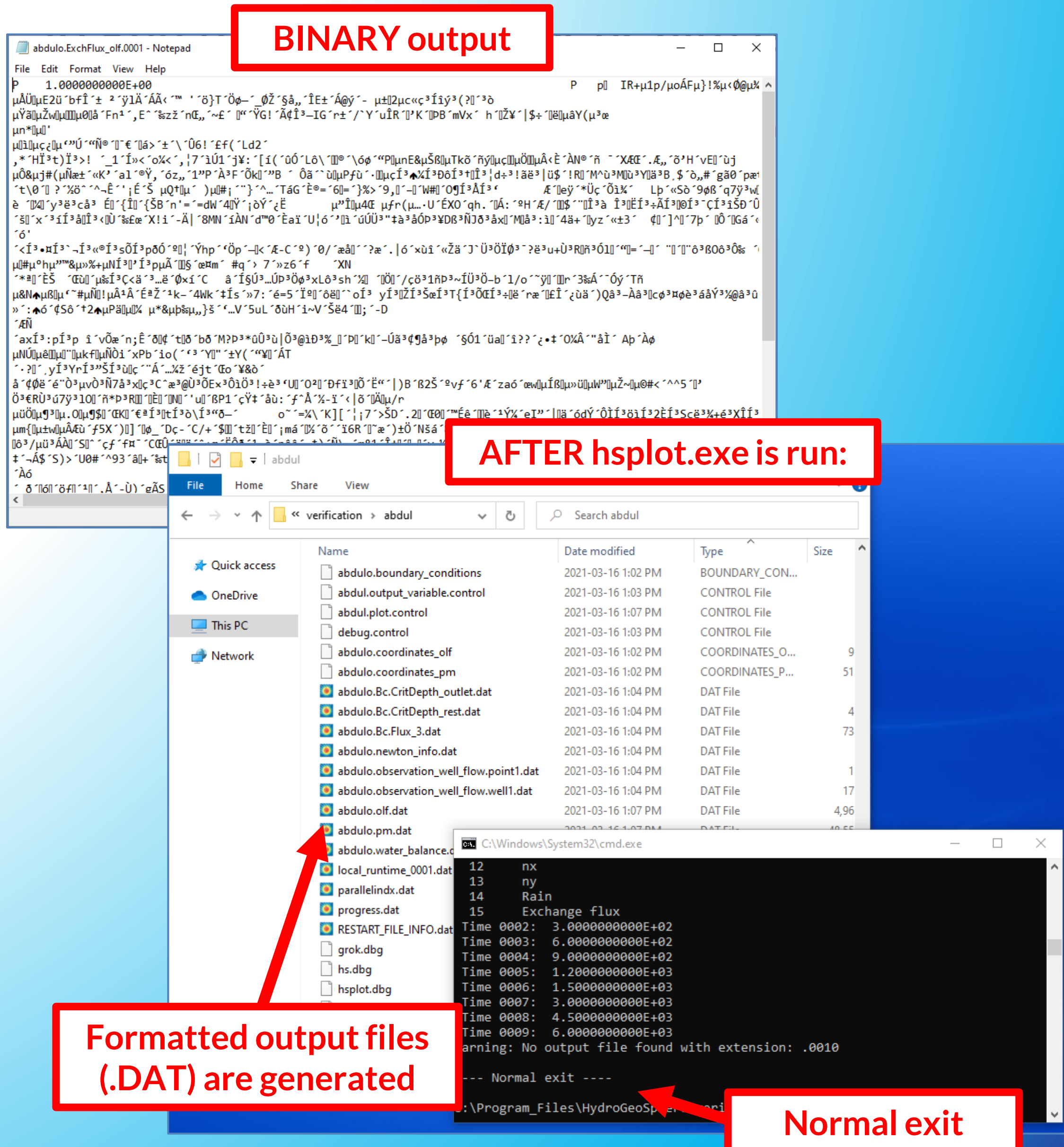
3) Run the phgs.exe executable

- *phgs.exe* is the program which solves the model
  - *i.e.* it performs the simulation
  - *i.e.* it solves the system of equations
- Command prompt (cmd) displays model run progress
- Use the command prompt (*cmd.exe*) to run *phgs.exe*
  - 1) type ‘*phgs.exe*’, press enter (↵)
  - 2) *phgs* progress is displayed
  - 3) *phgs* creates input files for visualization
  - 4) “---- normal exit----”



Run the hsplot.exe (or hs2vtu.exe) executabl

- *phgs.exe* output files are **binary**
  - *i.e.* you can't read/analyze them
- Results must be formatted for visualization and analysis using:
  - *hsplot.exe* (Tecplot)
  - *hs2vtu.exe* (ParaView, free)
- Use the command prompt (CMD) to run hsplot.exe
  - 1) type '*hsplot.exe*', press enter (↵)
  - 2) *hsplot* progress displayed
  - 3) "---- normal exit----"



Model Setup – *.grok, grok.exe

- Grok Philosophy
  - All aspects of model setup are controlled through the .grok file
  - Grok uses an intuitive scripting language for model setup (Chapter 5 of the user manual)
  - ! used to insert comments for documentation purposes

    e.g.,   ! This is a comment
  - A typical grok file is organized into major model components
    - Grid Generation
    - General Simulation Parameters
    - Material Definition
    - Initial and Boundary Conditions
    - Adaptive Time Step Controls
    - Output Control

Model Setup – *.grok, grok.exe

- **Problem Description**
  - **Can be any number of lines**
  - **Great for providing a detailed description of project, who was working on it, when, etc.,**
  - **“end title” used to indicate the end of the title section**

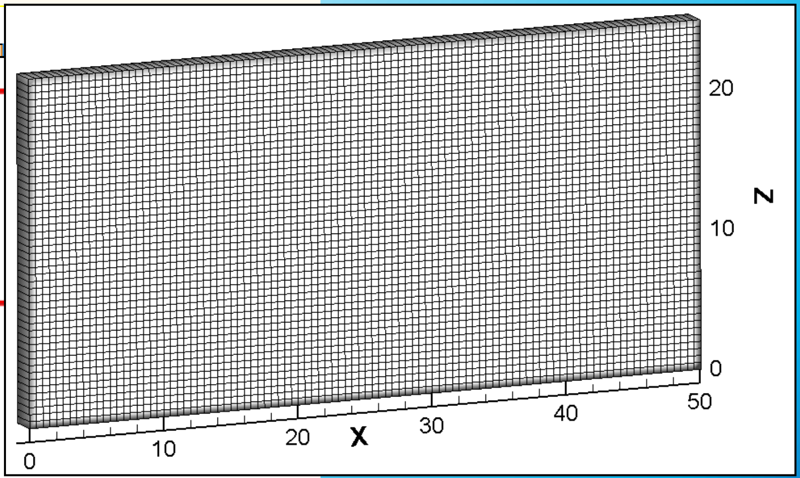
abdul.grok

```
!-------------------------  Problem description
Overland flow example 2
abdul, Monday, July 25, 2005 at 09:42
end title

!-------------------------  Grid generation
read gb 2d grid
./gb/grid

generate layers interactive
    zone by layer

    new layer
        layer name
        Base layer

        uniform sublayering
        5
```

Model Setup – *.grok, grok.exe

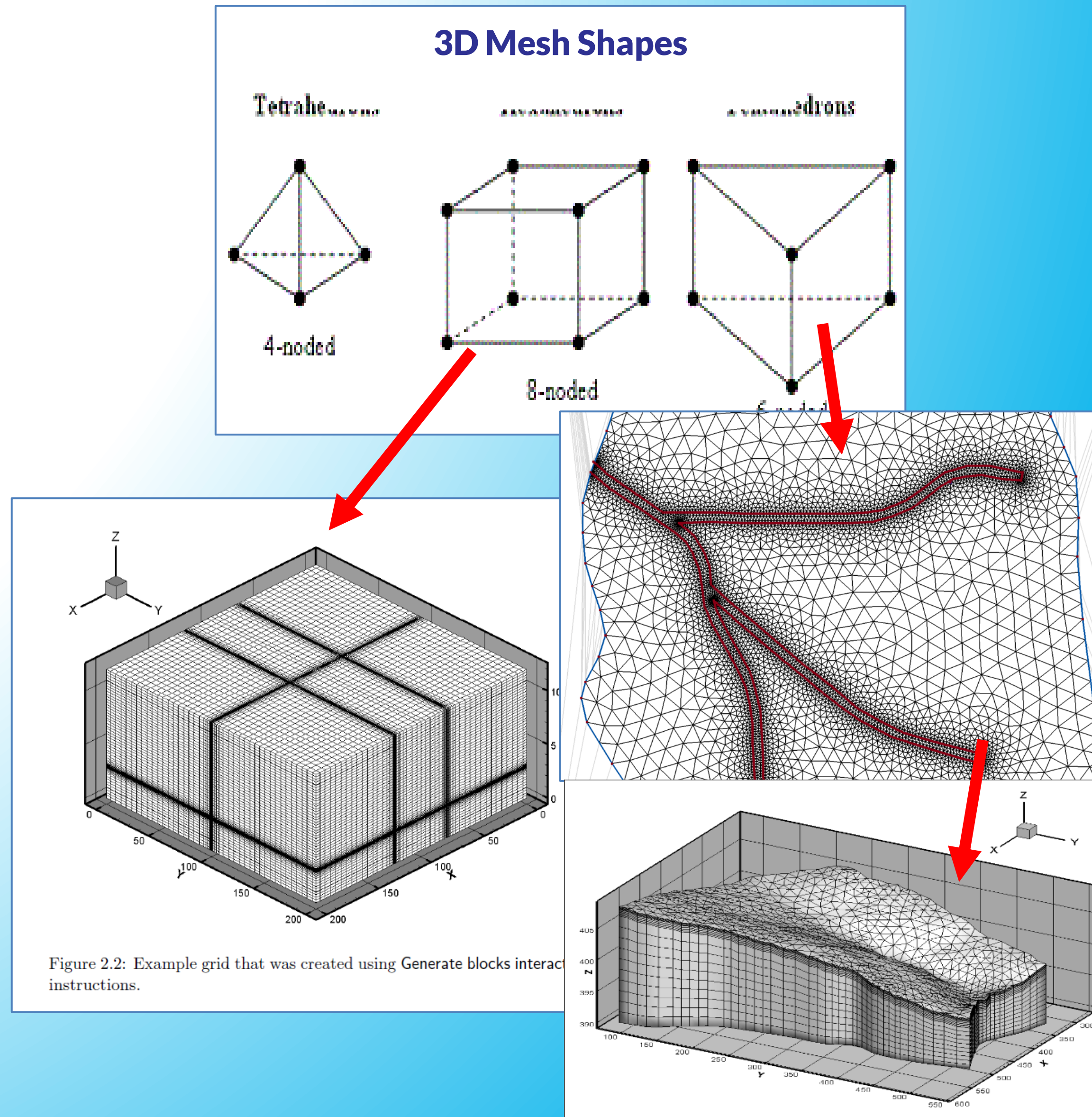
- **Grid Generation (Chapter 2.3 of Reference Manual)**
  - **Simple grid generation can be performed within grok**
  - **More complex grids (e.g., 3D triangular prism meshes) require 3rd party tools**

```
File  Edit  Search  View  Tools  Macros  Configure  Window  Help
Module1a.grok ×
!-------------------------- Problem description
Module 1a - Fully Saturated Confined Aquifer
Original Build Date: June 10, 2014 (S.Berg). Last Modified: Jul
end title
!-------------------------- Grid generation
generate uniform blocks
    50  100        ! Domain length and number of blocks in X
    1   1          ! Domain length and number of blocks in Y
    25  50         ! Domain length and number of blocks in Z
end grid generation

!-------------------------- General simulation parameters
units: kilogram-metre-second

!-------------------------- Porous Media Properties
use domain type
porous media

properties file
```



Control Volume Finite Elements (CVFE)

- **CVFE implementation** in HGS is very flexible, supporting a variety of grid types:
  - **Subsurface**: 8-node block or 6-node prism elements (3- and 4-node plate elements for fractures)
  - **Surface**: 3- and 4-node plate elements for surface water.
  - **Features**: 2-node line segments for wells, storm and sanitary sewers, water supply mains and tile drains or other types of linear infrastructure features.



Simple Hexahedral Grids

- Simple grids can be defined directly in the problem *.grok file:
  - Simple/uniform grid commands include:
    - Generate uniform blocks
    - Generate uniform prisms
  - Locally refined/variable grid commands include:
    - Generate variable blocks
    - Generate variable prisms
    - Generate blocks interactive
    - Grade x
    - Grade y
    - Grade z
  - Find information about these commands in the **HGS Reference Manual**
    - link in description

```
!---+----  Grid generation
generate blocks interactive
    grade x
        75.0 0.0 0.01 1.5 5.0
    grade x
        75.0 100.0 0.01 1.5 5.0
    grade x
        125.0 100.0 0.01 1.5 5.0
    grade x
        125.0 200.0 0.01 1.5 5.0
    grade y
        100.0 0.0 0.01 1.5 5.0
    grade y
        100.0 200.0 0.01 1.5 5.0
    grade z
        1.0 0.0 0.25 1.0 0.25
    grade z
        3.0 1.0 0.01 1.3 0.25
    grade z
        3.0 11.0 0.01 1.3 0.25
    grade z
        11.0 12.0 0.25 1.0 0.25
end generate blocks interactive
```

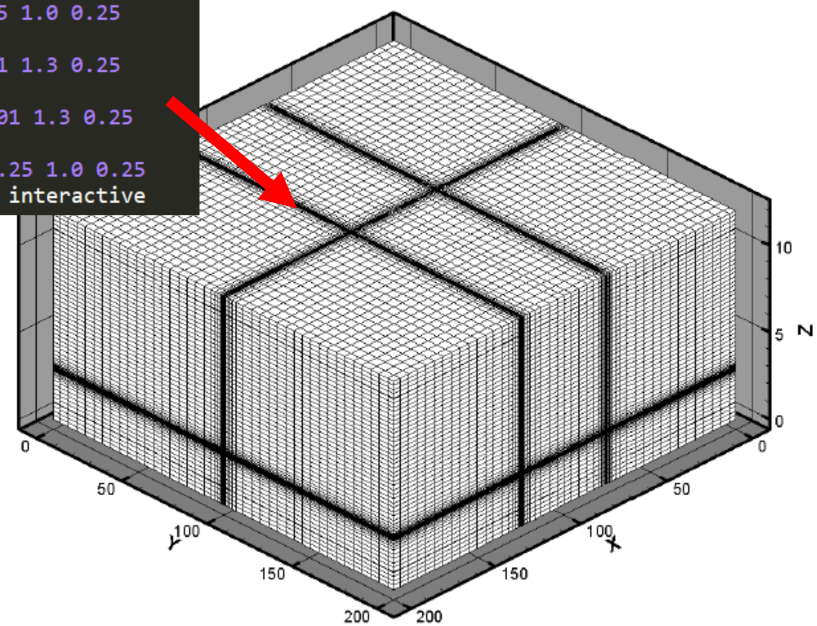
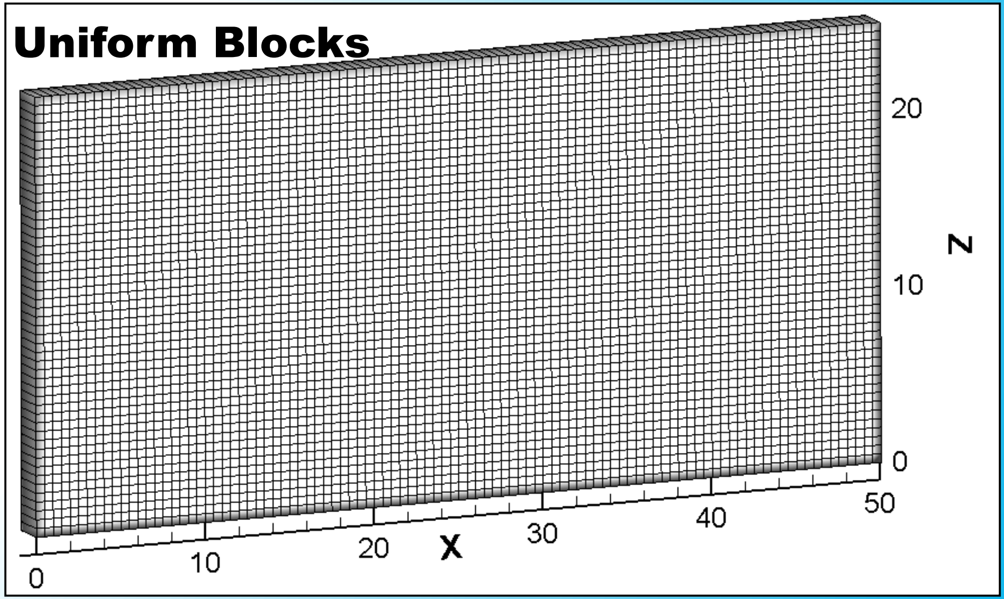
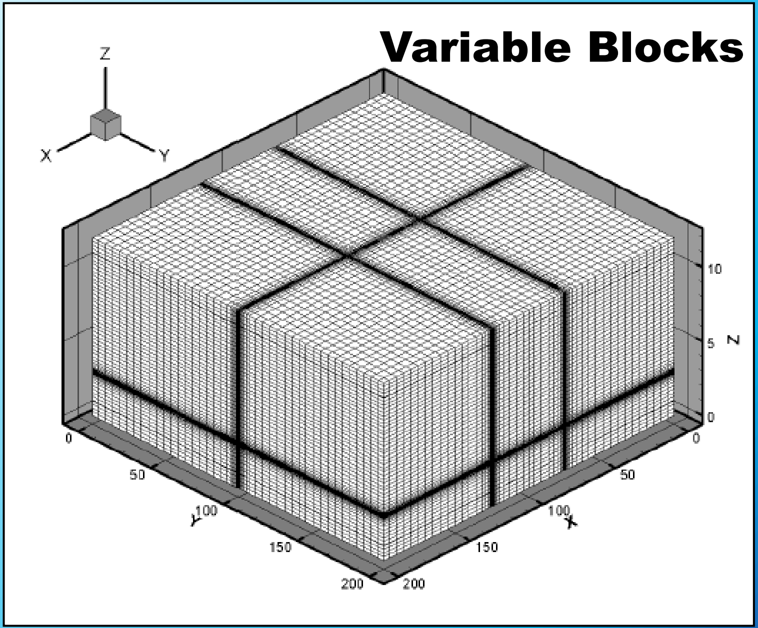


Figure 2.2: Example grid that was created using Generate blocks interactive instructions.

Model Setup – *.grok, grok.exe

- **Common Grid Generation Commands**
  - ***Generate Uniform Blocks***
    - **2D or 3D rectangular model with rectangular prism elements**
    - **Node spacing is constant in x, y, and z**
  - ***Generate Variable Blocks***
    - **2D or 3D rectangular model with rectangular prism elements**
    - **Node spacing is variable in x, y, and z**

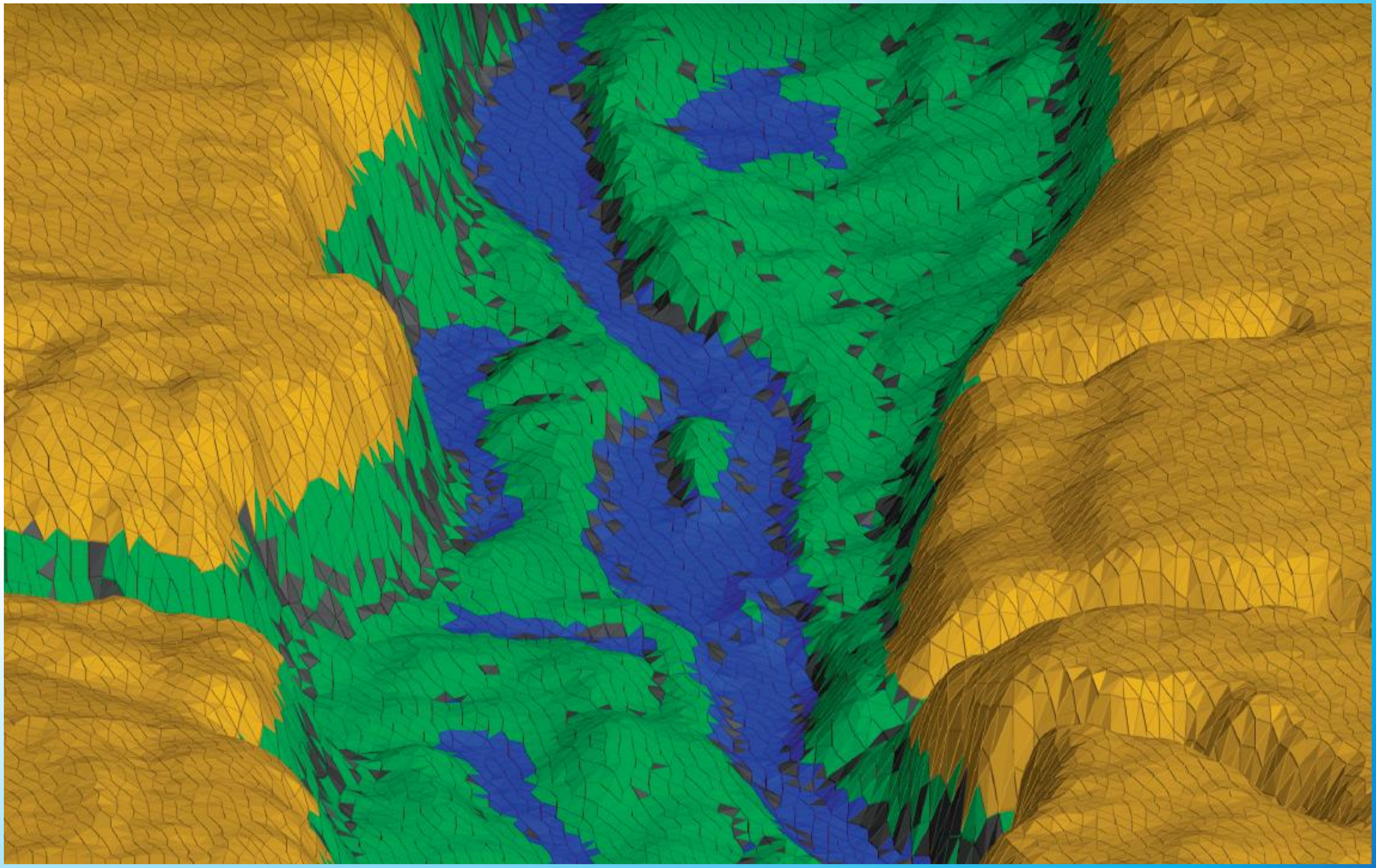




Model Setup – *.grok, grok.exe

▪ Grid Generation

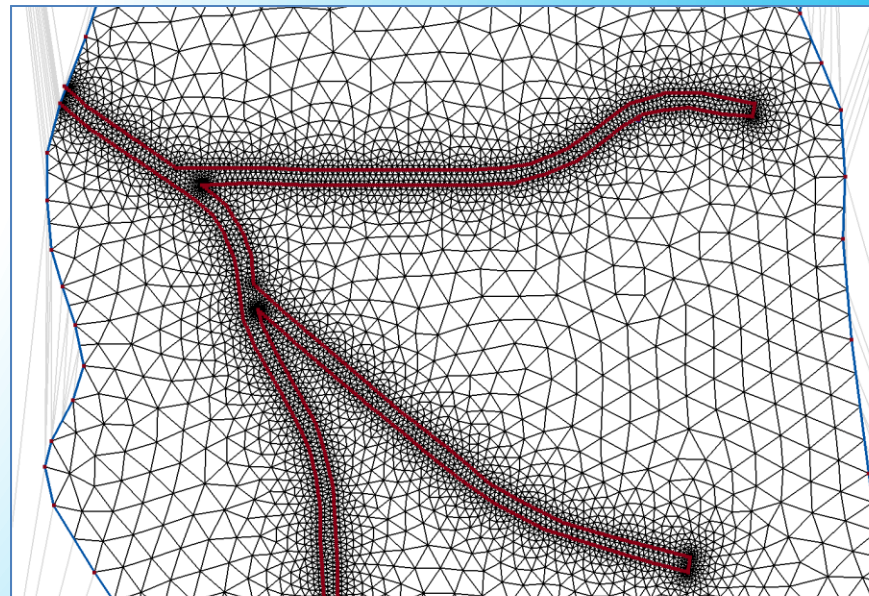
- More complex meshes, including selective mesh refinement and triangular prisms require 3rd party tools such as AlgoMesh or Leapfrog Hydro

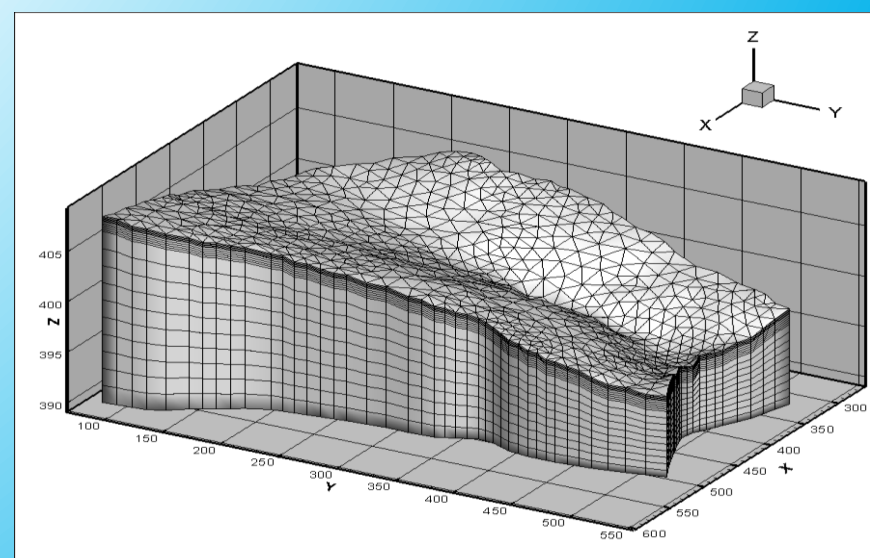


aquanty

Triangular Finite Element Meshes

- **Triangular meshes offer many benefits:**
  - Easily refine in areas of interest
  - Mesh conforms to boundaries
- **Aquanty recommends using AlgoMesh v2 for mesh generation:**
  - Fully integrated with HGS – export **.nchos and .echos files** (in addition to the 2D .AH2 mesh file)
  - Flexible geometry inputs – generate your mesh based on points, polylines and polygons from GIS or CAD data
  - Superior mesh generating algorithm to run models faster without sacrificing accuracy
  - www.hydroalgorithmics.com/software/algomesh
  - **Available for $2,000 USD ($1,000 USD for academics)**





AlgoMesh 2

HYDRO ALGORITHMICS

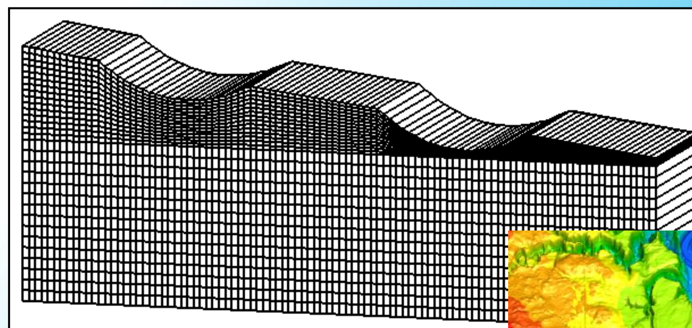
AlgoMesh tutorial

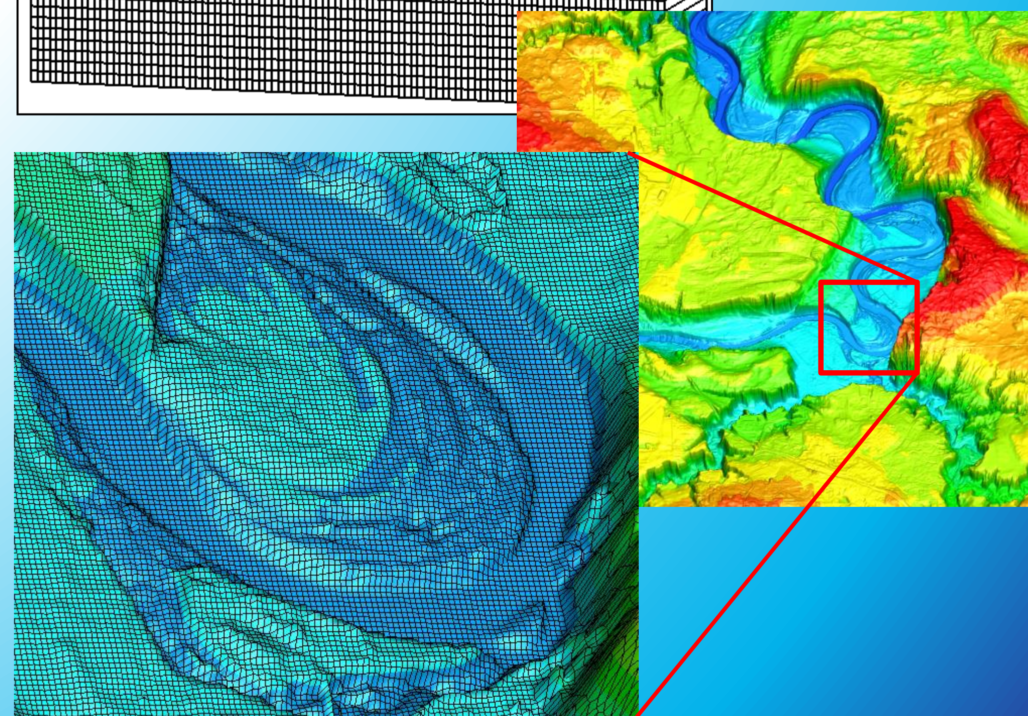
- https://community.aquanty.com/topic/597/algomesh-2d-mesh-generation-for-hgs
- This tutorial explores a number of techniques for producing a **high-quality triangular mesh** using AlgoMesh
- You will learn how to bring in a polygonal model extent boundary and roadway outlines, and manipulate these using AlgoMesh's built-in polyline editing and resampling tools.
- You will also learn how to export ***.nchos* and *.echos* files for HGS simulations**, and explore the **effects of different combinations of parameters** to AlgoMesh's mesh generation algorithms.

Model Setup – *.grok, grok.exe

- **Interactive 3D Mesh Generator**
  - **Define 2D Mesh (*Generate uniform rectangles*)**
  - **Define node layer positions to build 3D mesh (*Generate Layers interactive*)**

```
!-------------------------  Grid generation
generate uniform rectangles
    125 100
    1   1
    generate layers interactive

        base elevation
            elevation constant
            0
        end ! base elevation

        new layer
            layer name
            layer_2

            elevation constant
            19

            uniform sublayering
            15
        end

        new layer
            layer name
            layer_3

            elevation from xz pairs
                                        include ./2d_pothole_surf.txt
                end

            uniform sublayering
                        15
        end
    end ! generate layer interactive

end ! grid generation
```





aquanty

Model Setup – *.grok, grok.exe

- **General Simulation Parameters**
  - **Follows grid generation**
  - **This section of grok is used to define units to be used for the simulation (e.g., kilogram-metre-second)**
  - **By default the simulation is assumed to be fully saturated and steady state**
  - **Typical Commands**
    - ***Transient flow***
    - ***Unsaturated***
    - ***Do Transport***
  - **See Chapter 2.4 of the Reference Manual for more details**

Module3a.grok ×

```
!------------------------- Problem description
Module 3a - Variably Saturated - Transient to Steady State
Original Build Date: July 21, 2014 (S.Berg)
end title
!------------------------- Grid generation
generate uniform blocks
    25  250        ! Domain length and number of blocks in X
    1  1        ! Domain length and number of blocks in Y
    10  100            ! Domain length and number of blocks in Z
end grid generation

!------------------------- General simulation parameters
units: kilogram-metre-second
unsaturated
transient flow
```

Model Setup – *.grok, grok.exe

- **Selecting Mesh Components**
  - **Model setup requires assigning initial and boundary conditions and defining material properties**
  - **These actions are performed by selecting mesh components (e.g., nodes, elements, faces, segments, and zones)**
  - **Reference Manual (Chapter 2.4) contains an extensive list of commands for selecting Mesh Components**

Choose nodes block

1. **x1, x2** x-range of the block.
2. **y1, y2** y-range of the block.
3. **z1, z2** z-range of the block.

Choose nodes top boundary

Nodes around the top edge of the surface flow domain are chosen.

Choose nodes top

All nodes in the top sheet of the domain are chosen.

Choose elements x plane

1. **x1** The x-coordinate of the plane.
2. **ptol** Distance from the plane.

Model Setup – *.grok, grok.exe

- **Definition of Material Properties**
  - If not specified default values are assigned (listed in manual)
  - Material properties are specified using properties files
    - *.mprops – subsurface material properties (K, Ss, n, etc)
    - *.oprops – surface domain properties (Manning's friction, rill storage, etc)
    - *.dprops – dual continuum properties
    - *.fprops – discrete fracture properties
    - *.wprops – well properties
    - *.tprops – tile drain properties
    - *.cprops – channel properties
  - Chapter 2.8 of Reference Manual contains commands for assigning material properties
  - The *"scope"* section of the command tells you with what file the command can be used

K isotropic

`Scope: .grok .mprops`

1. **kval** Hydraulic conductivity [L T^{-1}].

Assign an isotropic hydraulic conductivity (i.e. $K_{xx} = K_{yy} = K_{zz}$).

• • •

Model Setup – *.grok, grok.exe

- **Assigning material properties**
  1. **Indicate which type of medium is being manipulated**

> Use domain type
>
> 1. **zone_type** Can be one of the strings: porous media, dual, fracture, surface, channel or et.

  2. **Indicate file name of properties file (e.g., abdul.mprops). This only needs to be done once within a grok file**
  3. **Select the zone to which you wish to assign material properties**
  4. **Choose material type from properties file**

Homogeneous Example

```
!------------------------- Porous media properties
use domain type                                          (1.)
porous media

properties file                                          (2.)
abdul.mprops

clear chosen zones                                       (3.)
choose zones all
read properties                                          (4.)
BORDEN SAND (FIELD SCALE) - DIGITIZED
```

Model Setup – *.grok, grok.exe

- **Assigning material properties**

Heterogeneous Example

Zone Assignment

```
!--------------------------  Zone creation
use domain type
porous media
choose elements block
0        125
0        1
16       30
choose elements block
0        125
0        1
5        13
new zone
1
clear chosen elements

use domain type
porous media
choose elements block
0        125
0        1
13       16
choose elements block
0        125
0        1
0        5
new zone
2
clear chosen elements
```

Material Assignment

```
!--------------------------  Porous media properties
use domain type
porous media

properties file
module5.mprops

clear chosen zones
choose zone number
1

read properties
BORDEN SAND (FIELD SCALE) - DIGITIZED

clear chosen zones
choose zone number
2

read properties
clay
clear chosen zones
```

Model Setup – *.mprops, *.oprops, etc.…

- **Properties File (*.mprops)**

```
! material
BORDEN SAND (FIELD SCALE) - DIGITIZED

    k anisotropic
    !0.36       0.36    0.036
    8.6 8.6 0.86

    porosity
    0.34

end ! material

! material
clay

    k anisotropic
    .001 .001 .0001

    porosity
    0.3

end ! material
```

Model Setup – *.grok, grok.exe

- **Initial Conditions (Chapter 5.6 of Reference Manual)**
  - **Need to be defined for subsurface, surface, and transport**
    1. **Choose nodes to which you wish to assign initial condition**
    2. **Assign initial head**

- **Subsurface Examples**
  - ***Initial head* - uniform value applied selected subsurface nodes**
  - ***Initial head surface elevation* – assigns an initial head of the surface elevation to all nodes at the same xy location**
  - ***Initial head from output file* - allows the simulation to be restarted with output from a previous simulation**

Model Setup – *.grok, grok.exe

- **Initial Conditions (Chapter 5.6 of Reference Manual)**
  - **Example from a coupled surface/subsurface flow model**

From Abdul Example

Subsurface

```
use domain type
porous media

choose nodes all
initial head
2.78
```

Surface

```
use domain type
surface

clear chosen nodes
choose nodes all
initial water depth
1.0e-4
```

Model Setup – *.grok, grok.exe

- **Boundary Conditions (Chapter 5.7 of Reference Manual)**
  - **Commonly Used Boundary Conditions**
    - ***Specified Head***
    - ***Specified Flux***
    - ***Rain***
    - ***Potential Evapotranspiration***
    - ***Critical Depth Boundary***
  - **Prior to assigning a boundary condition you must create either a node set, face set, or segment set as required by the specific boundary condition (documented in manual)**

- **Boundary Conditions (Chapter 5.7 of Reference Manual)**
  - **General Boundary Condition Structure**

```
boundary condition
    type
        {bc_type}

    node set/face set/segment set
        {bc_set_name}

    time value table/time raster table/time file table
        {bc_time(i), bc_file(i)...end}
            or
        {bc_time(i), bc_raster(i)...end}
            or
        {bc_time(i), bc_file(i)...end}

    constraints/tecplot options   !optional not required
end
```

Model Setup – *.grok, grok.exe

- **Boundary Conditions (Chapter 5.7 of Reference Manual)**
  - **Sample boundary condition setup (surface flux, e.g., rain)**

```
clear chosen nodes
choose nodes top

create face set
top

boundary condition
    type
    flux

    face set
    top

    time value table
    ! experiment 1: 2 cm/hr -> 5.555e-6 m/s for 3000 s (50 min)
    !
    !  time     flux
    0.0      5.555e-6
    3000.0   0.0
    end

    tecplot output

end ! flux bc
```

Model Setup – *.grok, grok.exe

- **Time step control**
  - This section of the grok file specifies all of the parameters related to time stepping during the HGS simulation
  - HGS can use fixed or adaptive time steps
  - Adaptive time steps provides the most accurate solution with the minimum amount of computational effort by adjusting to time step based on the behaviour of the problem
    - For example, if a large precipitation event may cause the time steps to become smaller to better simulate the rapidly changing system.
    - During periods of little change, the time steps become larger because the system is more stable
  - Adaptive time stepping can be influenced by the behaviour of the following physical parameters
    - Hydraulic head, surface water depth, soil saturation, solute concentration
  - The numerical behaviour of the solution can also influence the adaptive time stepping (e.g., Newton Iterations)

Model Setup – *.grok, grok.exe

- Time step control
  - The following minimum information should be specified for a transient simulation (if not specified defaults will be used)
    - *Initial time* – the start time of the simulation (e.g., 0)
    - *Initial time step* – the size of the first attempted time step (e.g., 0.01)
    - *Maximum timestep multiplier* – based on the behaviour of the simulation HGS will multiply the previous timestep, this sets a maximum increase per timestep (e.g. 2x)
    - *Output times* – Specifies the times at which a snapshot of the simulation is output. Last value controls when the simulation is completed

Model Setup – *.grok, grok.exe

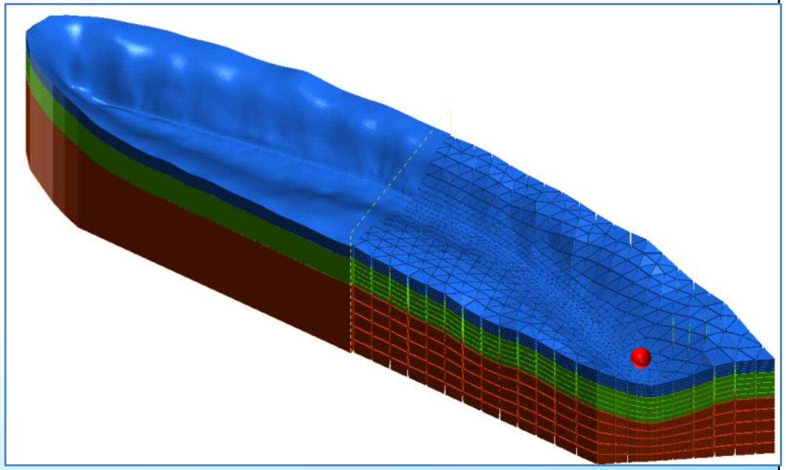
```
!-------------------------  Timestep controls
head control
0.5

saturation control
0.050

newton iteration control
10

maximum timestep
100.

initial timestep
0.5

maximum timestep multiplier
2.0

minimum timestep multiplier
0.5

output times
   1
   300.0
   600.0
   900.0
  1200.0
  1500.0
  3000.0
  4500.0
  6000.0
end
```

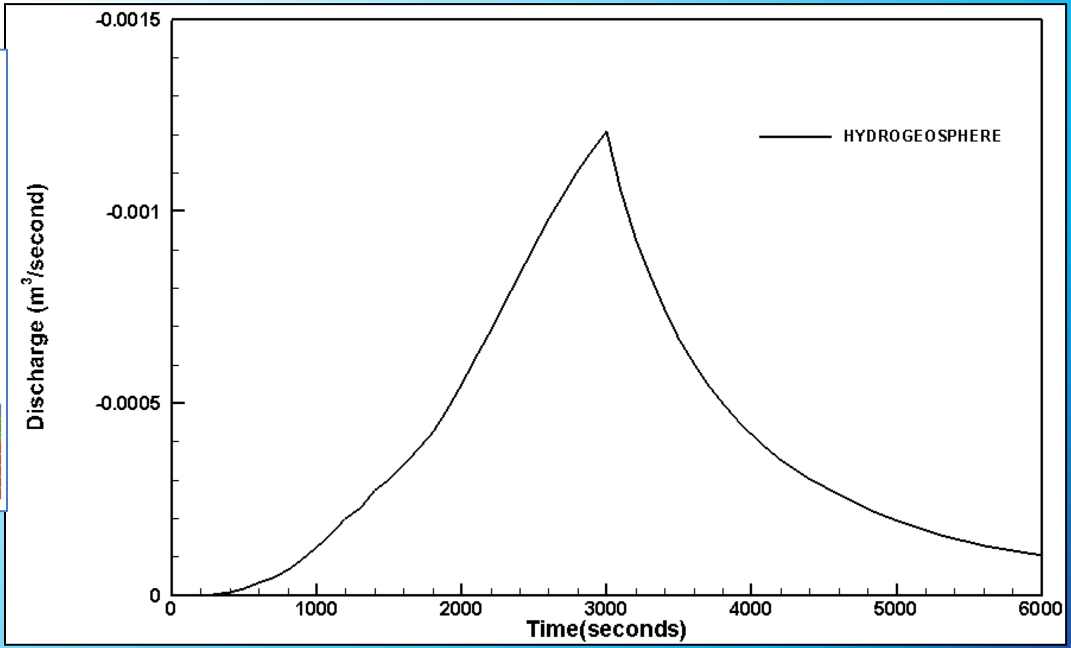
```
 1. Initial time =   0.000000000000000E+000

----------------------abdul Step:       1-------------------------
Global target time:          1.00000(     1 of      9)
 %done       Time                delta_t            Tnext
  0.00         0.00000            0.50000            0.50000 Accept timestep
     Calculating transient flow solution...

Summary of nonlinear iteration
Iter Relfac          Delval   @Node NcNode          Resval   @Node  NcNode Solv   Dom
   0(Initial)     0.000                         1.64371E-05  23272       0
   1   1.000     5.2899E-03   20614       0  -3.23633E-07  23276       0    12  pm,of
 CONVERGENCE: delval=  0.52899D-02 <  0.10000D+00
              resval = 0.32363D-06 <  0.10000D+00


 Variable                 Max. change      Target change    Dt multiplier    At node
 ========                 ===========      =============    =============    =======
 Head                     0.52899E-02        0.50000            94.520        20614
 Water depth              0.12991E-04        1000.0           0.76976E+08     22846
 Saturation               0.21939E-04        0.50000E-01        2279.0        20614
 NR Iteration               1.0000           10.000             10.000            0
  Timestep multiplier:     10.0000000000000

  Minimum dt_multiplier    >     Maximum allowed
   10.0000000000000        >     2.00000000000000
  Timestep multiplier:     2.00000000000000
 Accepted solution at time    0.500000000000000
```

Model Setup – *.grok, grok.exe

- **Output Control (Chapter 5.9.2/3)**
  - Many options for recording the details of a simulation at a point (e.g., monitoring well, surface hydrograph)
  - A time series of every parameter in the simulation is written for each observation location

Stream Flow at Domain Outlet





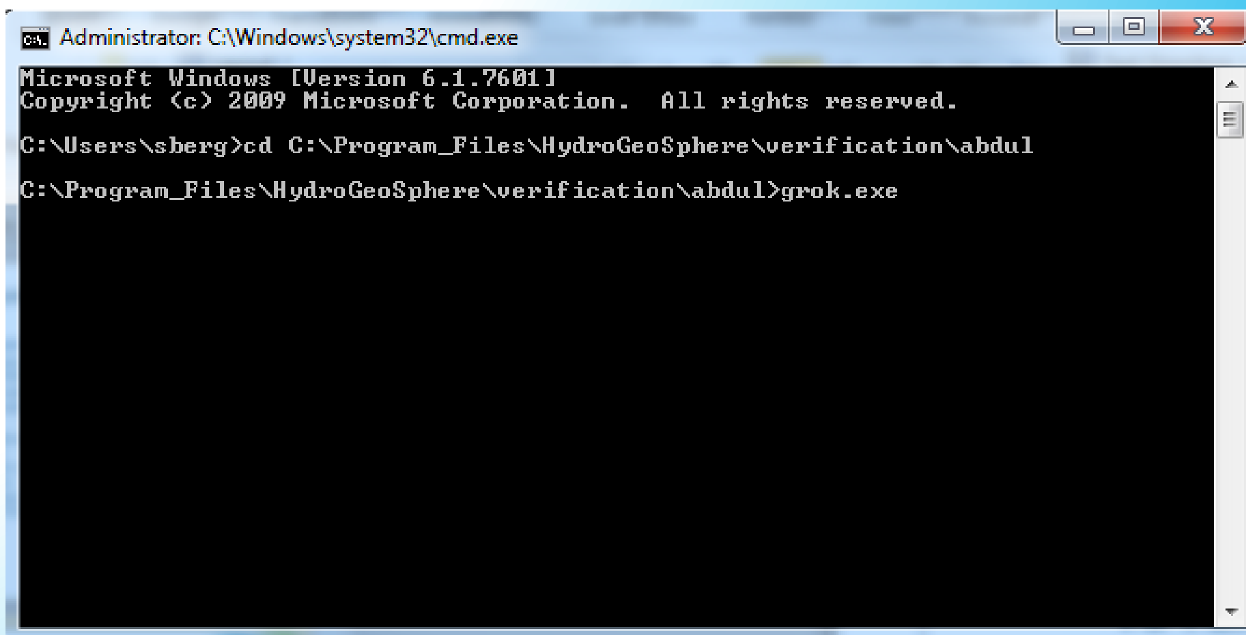
Model Setup – *.grok, grok.exe

- **Grok Summary**
  - All aspects of model setup are controlled through the .grok file
  - Grok uses an intuitive scripting language for model setup (Chapter 5 of the user manual)
  - ! used to insert comments for documentation purposes

    e.g., ! This is a comment
  - A typical grok file is organized into several major sections
    - Grid Generation
    - General Simulation Parameters
    - Material Definition
    - Initial and Boundary Conditions
    - Adaptive Time Step Controls
    - Output Control
  - Grok processes all of the commands and data into a format that phgs.exe will recognize for the simulation

Model Setup – *.grok, grok.exe

- **Grok Summary**
  - **To run grok**
    - **Copy grok.exe into simulation folder (if grok.exe isn`t set in path)**
    - **Launch command prompt and navigate to folder**



    - **Run grok.exe**

What Do the Executables Do?

- HGS consists of 3 Key Executables
  1. grok.exe – Compiles *.grok file which contains the model definition and setup information. Prepares all information for HGS

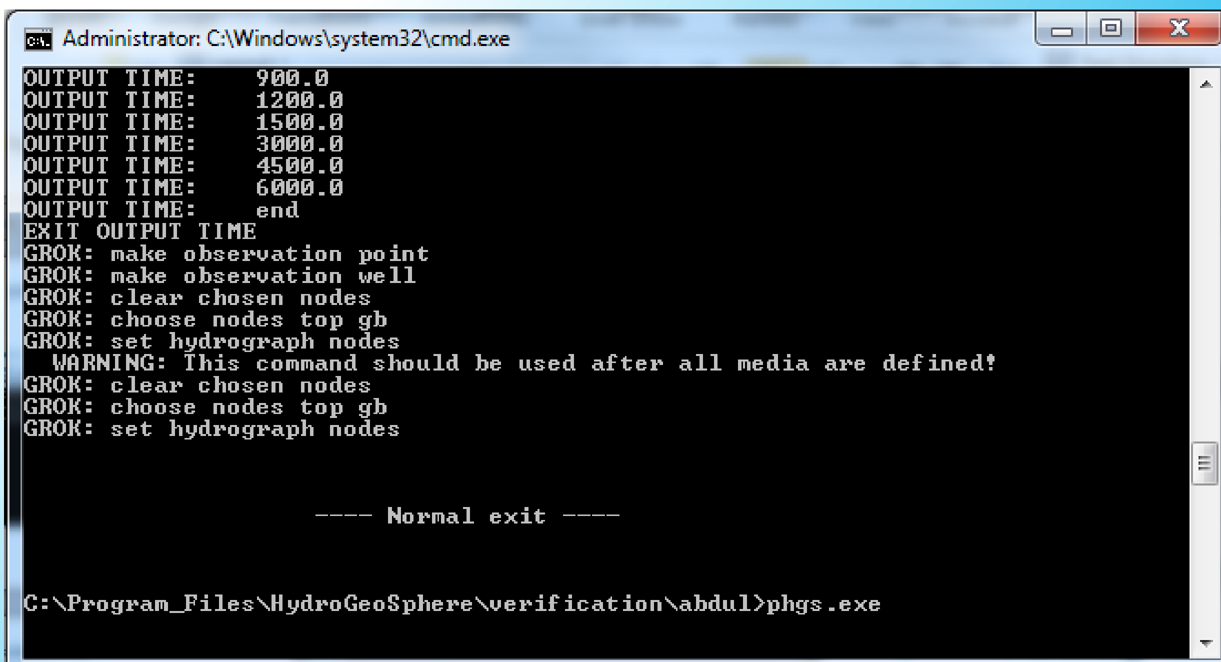
     (Chapter 5 commands are used to setup grok files)

  2. phgs.exe – Serial or parallel numerical simulation

  3. hsplot.exe – Processes model output into a format readable by 3D visualization products (e.g., Tecplot)

Running a Model – phgs.exe

- **Following the successful execution of grok.exe you are now ready to run phgs.exe**
  - **To run phgs**
    - **Copy phgs.exe into simulation folder (if phgs.exe isn`t set in path)**
    - **Launch command prompt and navigate to folder**
    - **Run phgs.exe**



Parallel Computing with HGS

- HGS is parallelized using the OpenMP shared memory framework.
- HGS can take advantage of up to 16 available CPU threads on a single machine
- HGS will not work across multiple machines on a cluster
- Level of parallelization is set using the Parallelindx.dat file
- Parallelindx.dat is created within the simulation folder when HGS.exe is first ran, can be copied in from another folder.

- Serial by default
- First 2 values should be the same
- Solver Type = 1 ➔ Serial Mode
- Solver Type = 2 ➔ Parallel Mode
- Can usually be ignored: used for model restart (in case of crash)

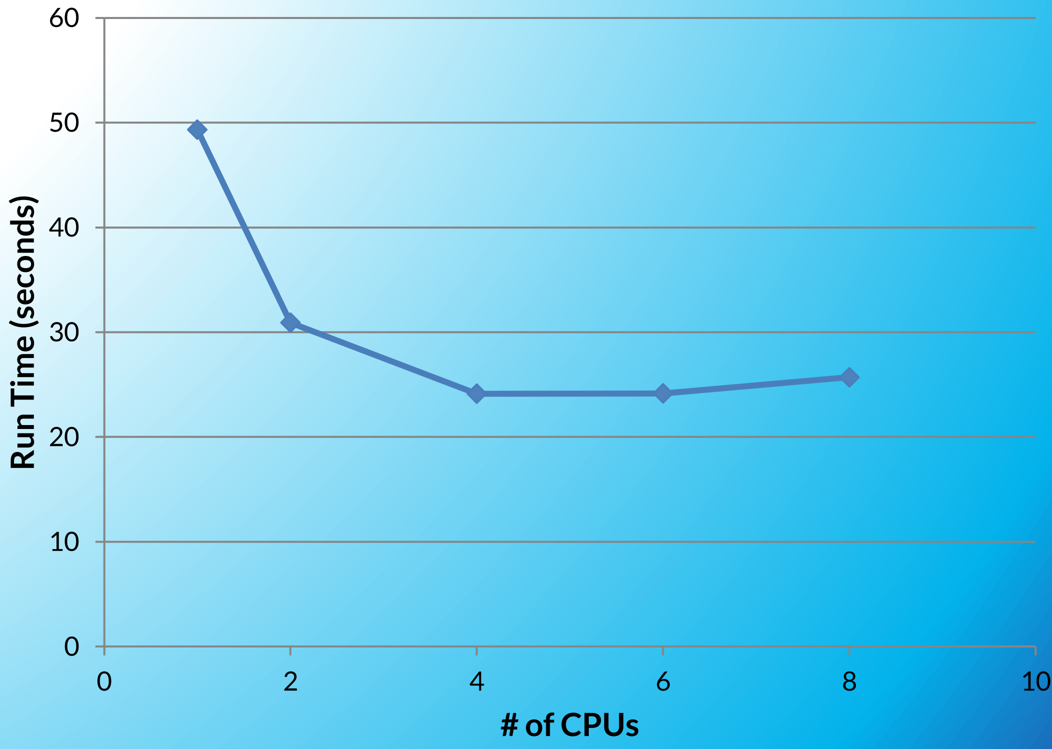
parallelindx.dat

```
__Number_of_CPU
         1
__Num_Domain_Partitiong
         1
__Solver_Type
         1
__Coloring_Input
 F
__Wrting_Output_Time
 -1.00000000000000
__Simulation_Restart
         1
```

Parallel Computing with HGS

- **Optimal number of CPU threads is usually around 1 thread per 100,000 nodes, however this is problem dependent**
- **Each problem will experience a performance plateau where using additional threads does not result in a decrease in simulation time**

Graph is for the Abdul problem. Approximately 22,000 nodes



Running a Model – phgs.exe

- **Understanding what the model is displaying**

```
-----------------------abdul Step:       11------------------------
Global target time:         300.00000(      2 of       9)
 %done       Time            delta_t           Tnext
  3.57        214.00000        100.00000         314.00000 !! Tnext >= target
  3.57        214.00000         86.00000         300.00000 1 step to target
     Calculating transient flow solution...

Summary of nonlinear iteration
Iter Relfac          Delval  @Node NcNode        Resval  @Node NcNode Solv   Dom
   0(Initial)     0.000                     1.70025E-05  21900      0
   1  1.000      0.1243       21522      37  -3.55709E-06  20594      0    25 pm,pm
 Failed nodal flow check
   2  1.000      5.7765E-02   21897       0   4.88870E-07  22285      0    23 pm,of
 CONVERGENCE: delval=  0.57765D-01 <  0.10000D+00
              resval = 0.48887D-06 <  0.10000D+00


 Variable             Max. change      Target change   Dt multiplier   At node
 ========             ===========      =============   =============   =======
 Head                 0.15655           0.50000           3.1939        20913
 Water depth          0.55828E-03        1000.0          0.17912E+07    23013
 Saturation           0.39818E-01       0.50000E-01       1.2557        21711
 NR Iteration          2.0000            10.000           5.0000            0
  Timestep multiplier:     1.25572547454452
 Accepted solution at time     300.000000000000

  Writing output to files at time     300.000000000000
 Saving saturations
 Saving heads
 Saving heads overland flow
 Saving exchange flux between surface/subsurface
 Saving velocities
 Saving darcy fluxes
 Saving overland velocities
 Met global target time     300.000000000000
 Restore last big timestep used    100.000000000000
```

Running a Model – phgs.exe

- Real-time control of simulation behaviour (debug.control)
- When phgs.exe is started it creates a file called debug.control
- This file contains all of the convergence parameter and adaptive time step control parameters
- These can be modified during the simulation to influence simulation behaviour

- Look at sample debug.control file

aquanty

Running a Model – phgs.exe

- **Simulation Time Report – Generated at the end of the simulation**

```
---------- SIMULATION TIME REPORT
number of CPU applied                =          4
number of time step                  =         68
number of total Newton-Ralphson loop =         82
number of total solver iteration     =       2527
number of equation                   =      23324
Global assembly  time     =   10.976
ILUC             time     =  0.23206
Solver           time     =   1.2466
SetBC            time     =  0.61868E-01
PARTIAL simulation time   =   23.940

Time of operation was    76.175     seconds

                    ---- NORMAL EXIT ----


C:\Program_Files\HydroGeoSphere\verification\abdul>
```

Visualization of Simulation Results
hsplot.exe

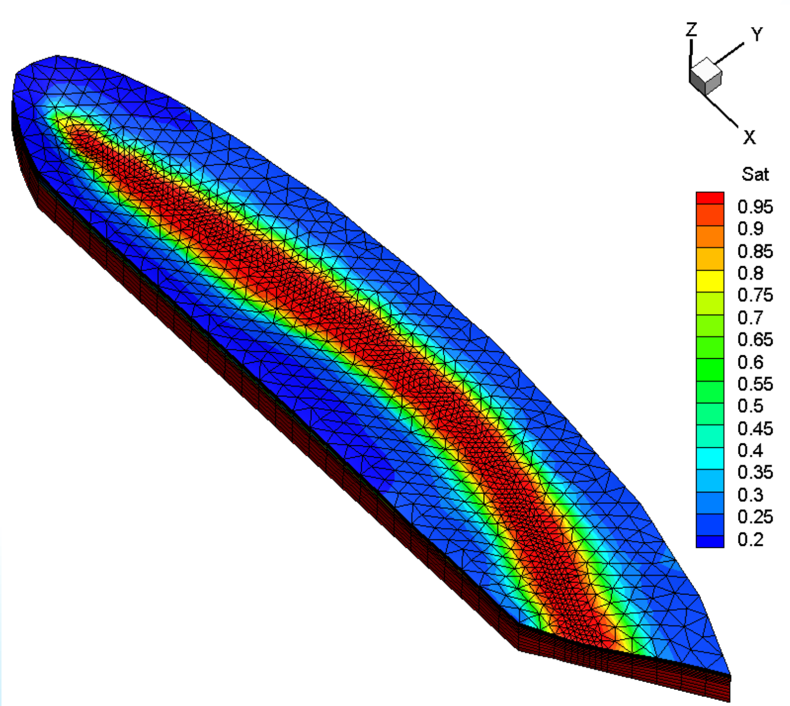
- **Following completion of phgs.exe, or after a snapshot has been written, hsplot.exe converts the output binaries into Tecplot format**
  - **To run hsplot**
    - **Copy hsplot.exe into simulation folder (if hsplot.exe isn`t set in path)**
    - **Launch command prompt and navigate to folder**
    - **Run hsplot.exe**

```
olf data to tecplot
Creating tecplot file abdulo.olf.dat

   1      X
   2      Y
   3      Z
   4      Zone
   5      Head
   6      Depth
   7      LogDepth
   8      Vx
   9      Vy
  10      Vz
  11      Exchange flux
  12      3DNode#
 Time 002:          300.00000
 Time 003:          600.00000
 Time 004:          900.00000
 Time 005:         1200.00000
 Time 006:         1500.00000
 Time 007:         3000.00000
 Time 008:         4500.00000
 Time 009:         6000.00000

---- Normal exit ----

C:\Program_Files\HydroGeoSphere\verification\abdul>
```

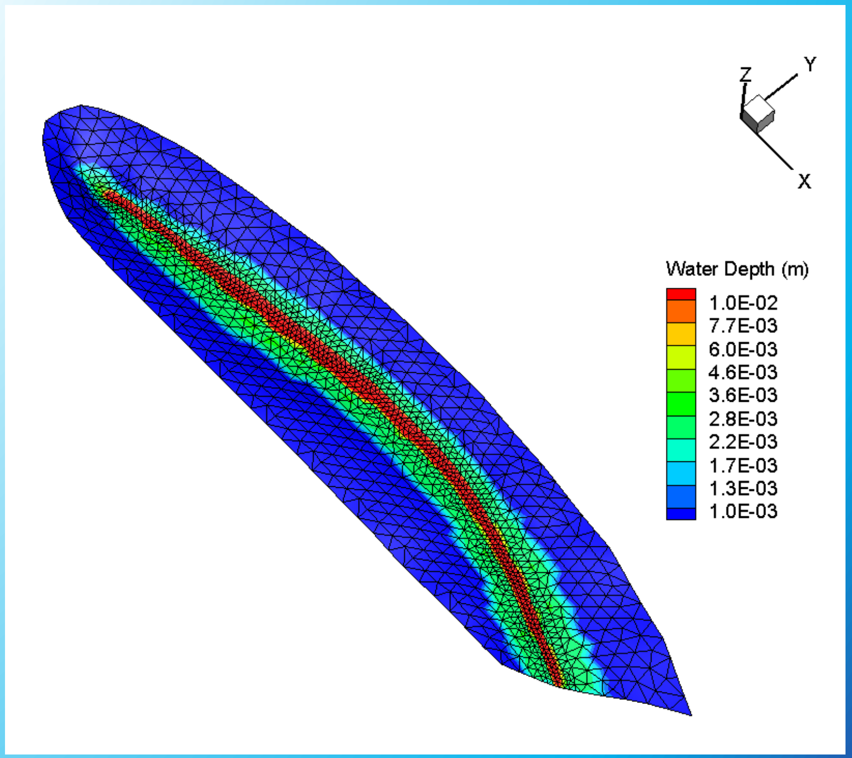
Visualization of Simulation Results hsplot.exe

- *o.pm.dat – porous medium simulation results
- *o.olf.dat – overland flow simulation results

Subsurface Saturation



Surface Water Depth



Hands-on Examples

- **Things to try….**
  - **Explore the grok files in C:\Program_Files\HydroGeoSphere\Verificaton**
  - **Try running some problems**
    - **→Grok.exe → phgs.exe → hsplot.exe**
  - **Play with different levels of parallelization (within the limits of your machine)**
  - **View the results in Tecplot**

Hands-on Examples

- Things to try….
  - You can also explore the 'Introductory Modules' example problems, a series of increasingly more complex box models:
    - Module 1a – Fully saturated, homogeneous, steady-state flow
    - Module 1b – Saturated, homogeneous, transient transport
    - Module 2a – Saturated, heterogeneous, steady-state flow
    - Module 2b – Saturated, heterogeneous, transient transport
    - Module 3a – Variably saturated, homogeneous, transient
    - Module 3b – Variably saturated, homogeneous, transient with recharge
    - Module 4a – Saturated, homogeneous, steady-state, discrete fractures
      - Module 4b – Saturated, homogeneous, transport with discrete fractures
    - Module 4c – Saturated, homogeneous, steady-state, random fractures
    - Module 4d – Saturated, homogeneous, transport with random fractures

Model Details

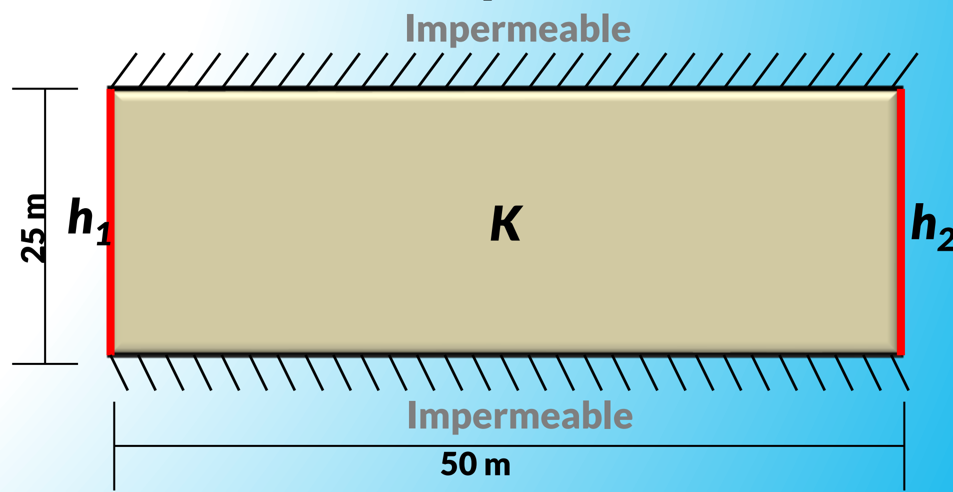
Node Count = 10,302

K = 1 x 10^{-5} m/s

$h1$ = 50 m

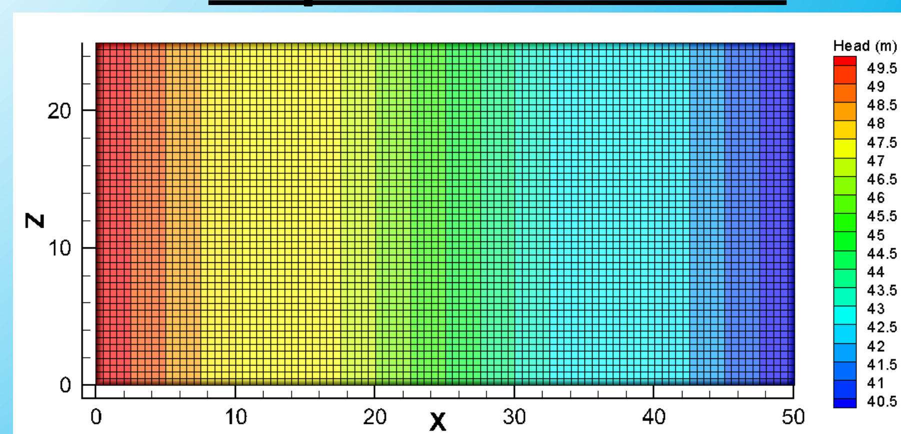
$h2$ = 40 m

Conceptual Model



Things to try:
- Change boundary condition values
- Change hydraulic conductivity
  - (look at impact on boundary flux (*.lst))
- Change the extent of the boundary condition
- Time varying boundary condition

Sample Simulation Results



Module 1b

Fully Saturated | Homogeneous | Steady Flow | Transient Transport

Model Details

Node Count = 10,302

K = 1 x 10^{-5} m/s

h1 = 50 m

h2 = 40 m

Source x, z = 5, 12.5 m

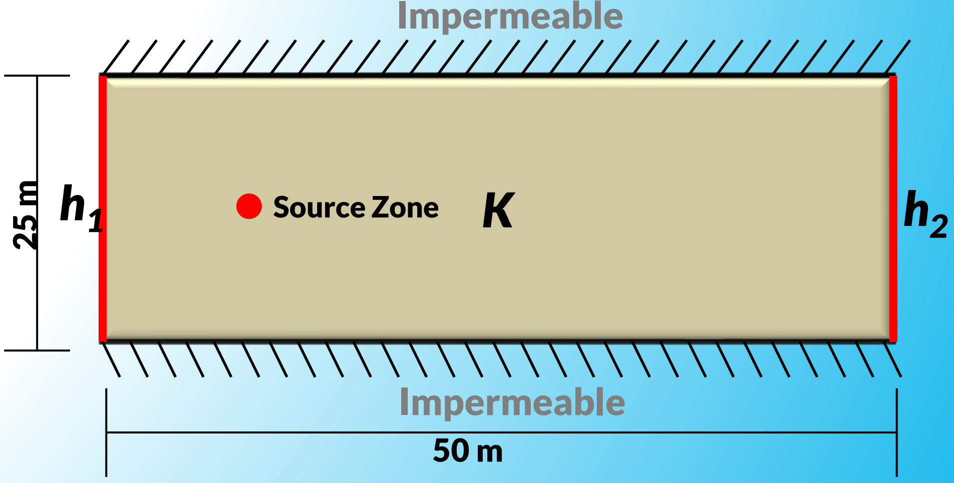
Source Concentration = 1 kg/m^3

Output Times = 1, 5, 10, 20, 30, 50, 60, 80, 100 days

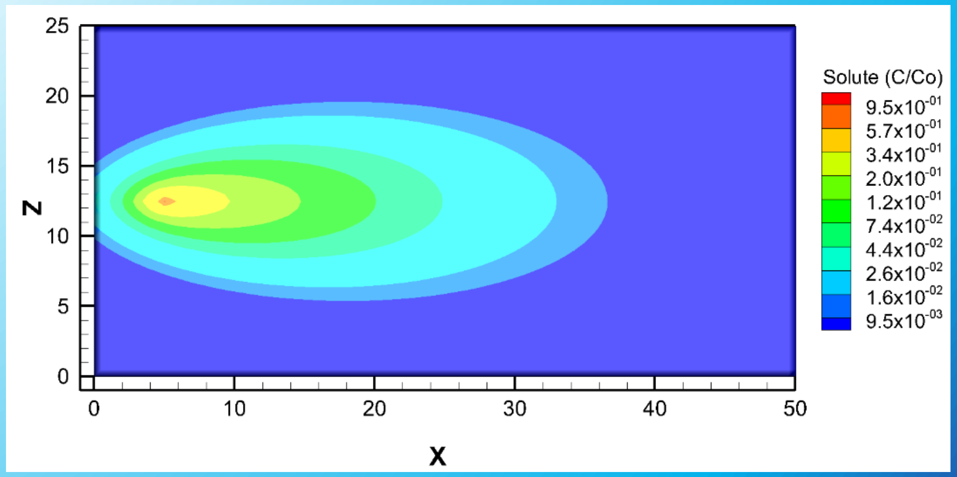
Things to try:

- Change source location
- Change constant head values
- Change source concentration
- Adjust mesh refinement

Conceptual Model



Sample Simulation Results



Model Details

Node Count = 10,302

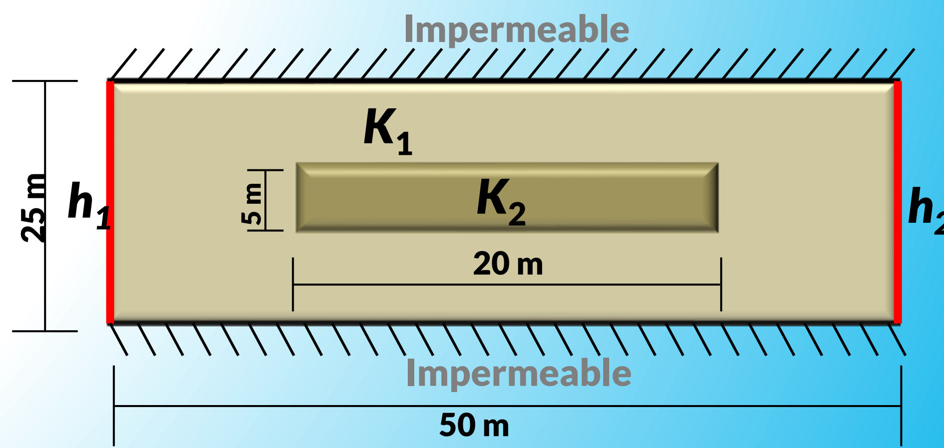
K 1= 1 x 10^{-4} m/s

K 2= 1 x 10^{-6} m/s

h1 = 50 m

h2 = 40 m

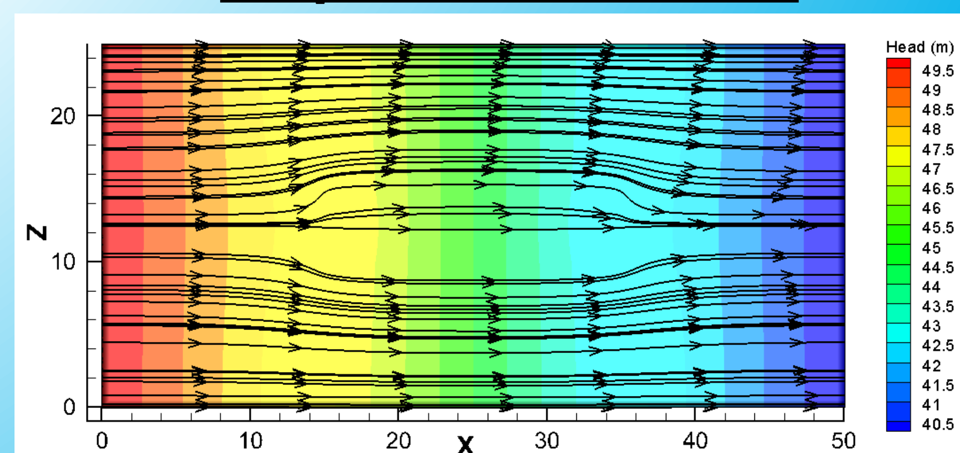
Conceptual Model



Things to try:

- Change boundary condition values
- Change hydraulic conductivity
    - (look at impact on boundary flux (*.lst))
- Change the extent of the boundary condition
- Time varying boundary condition

Sample Simulation Results



Model Details

Node Count = 10,302

K 1= 1 x 10^{-4} m/s

K 2= 1 x 10^{-6} m/s

h1 = 50 m

h2 = 40 m

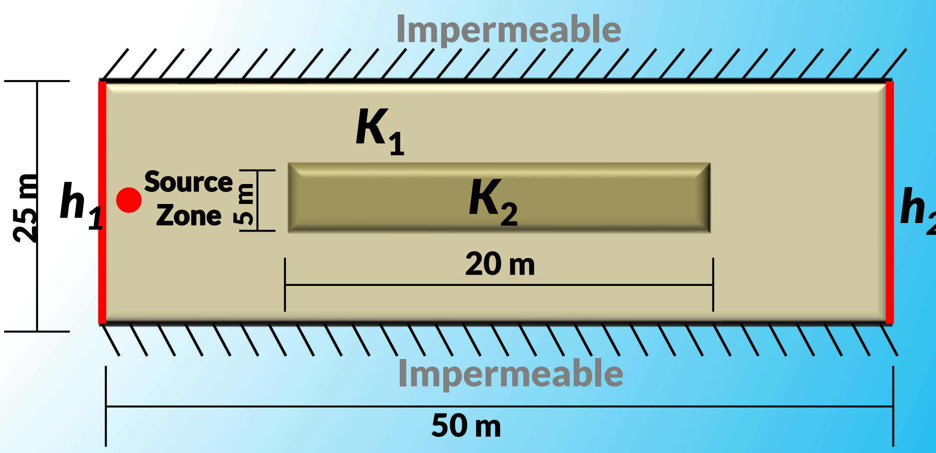
Source *x*, *z* = 5, 12.5 m

Source Concentration = 1 kg/m^3

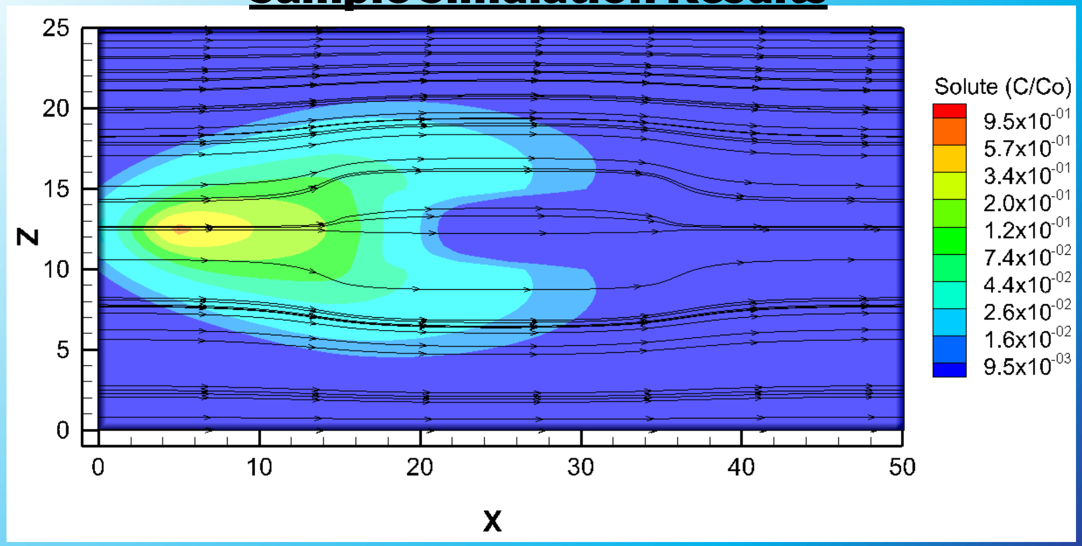
Things to try:

- Change source location
- Change constant head values
- Change hydraulic conductivity

Conceptual Model



Sample Simulation Results



Module 3a

Variably Saturated	Homogeneous	Transient to Steady State		

Model Details

Node Count = 50,702

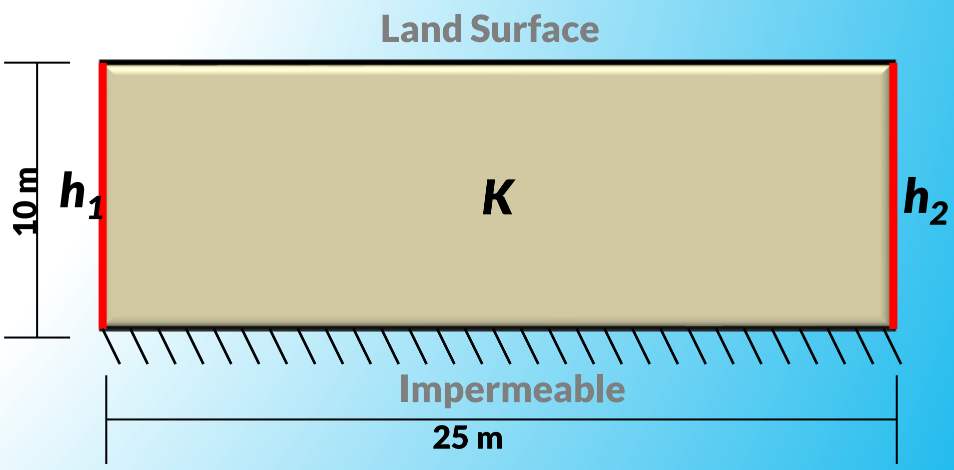
K = 1 x 10-5 m/s

h1 = 8 m

h2 = 5 m

Soil-water retention curves (Borden Sand)

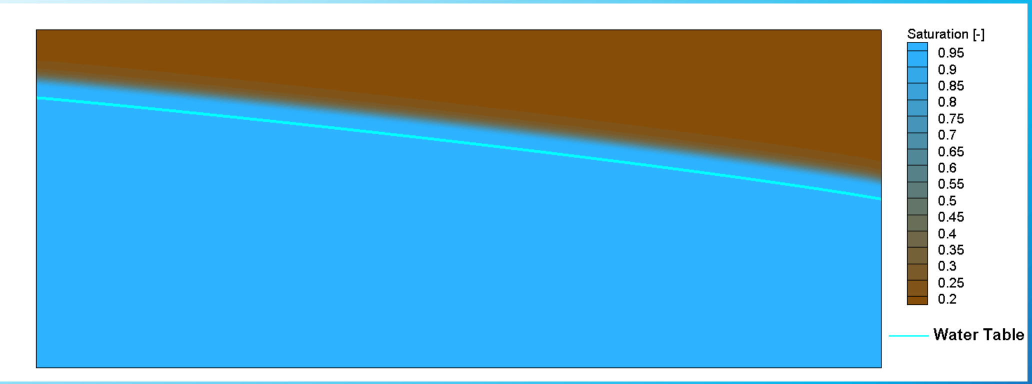
Conceptual Model



Things to try:

- Change constant head values
- Change hydraulic conductivity
- Adjust level of parallelization
- Adjust soil water retention curves
  (see spreadsheet included)

Sample Simulation Results



Module 3b

Variably Saturated	Homogeneous	Transient to Steady State		

Model Details

Node Count = 50,702

K = 1 x 10^{-5} m/s

h1 = 8 m

h2 = 5 m

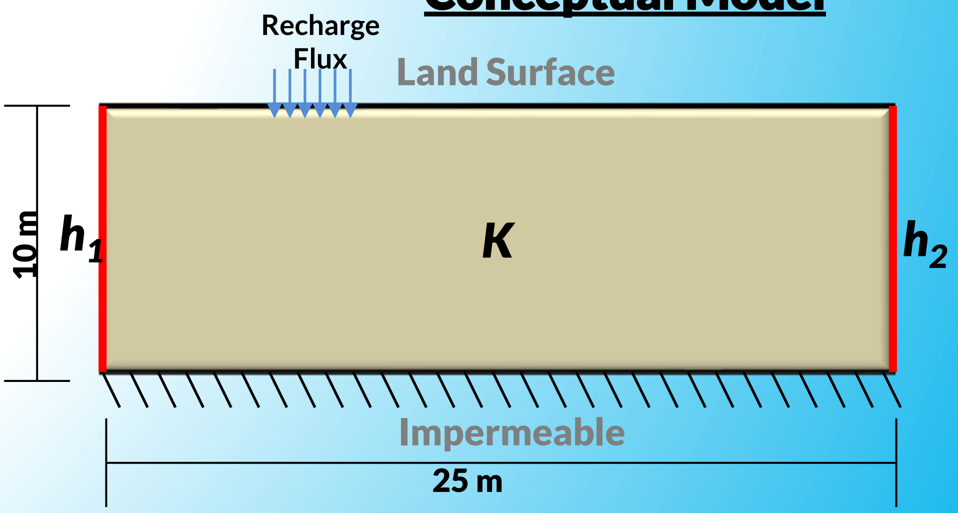
Recharge Flux = m/s

Soil-water retention curves

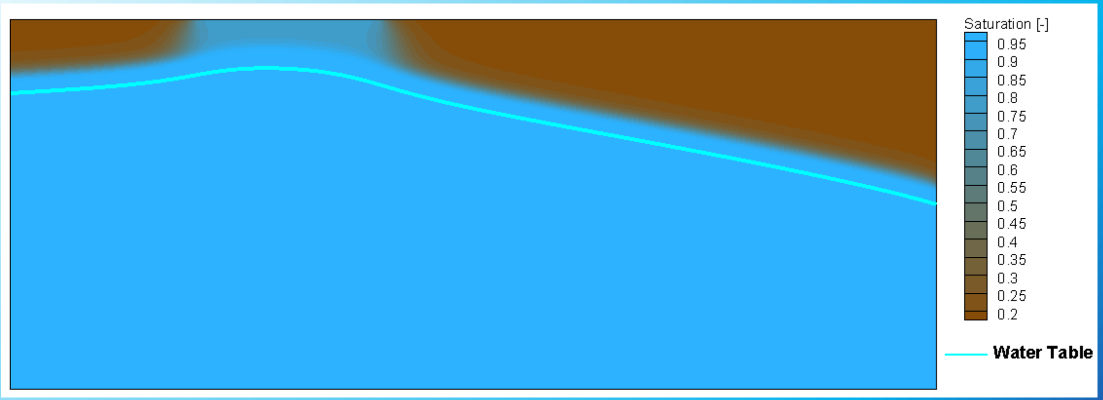
Things to try:

- Change recharge rate
- Change recharge location
- Adjust soil water retention curves
  (see spreadsheet included)
- Change hydraulic conductivity

Conceptual Model



Sample Simulation Results



Model Details

Node Count = 10,302

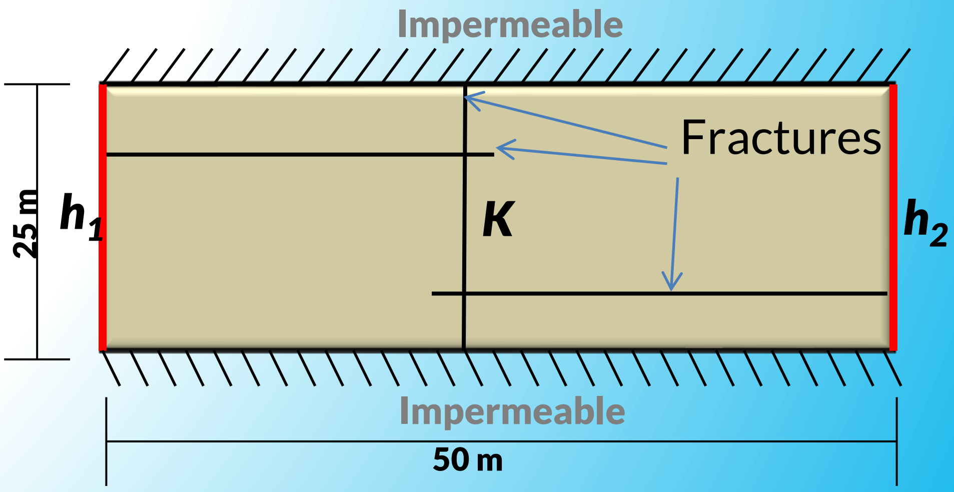
K = 1 x 10^{-9} m/s

h1 = 50 m

h2 = 40 m

Fracture Aperture = 1 x 10^{-4} m

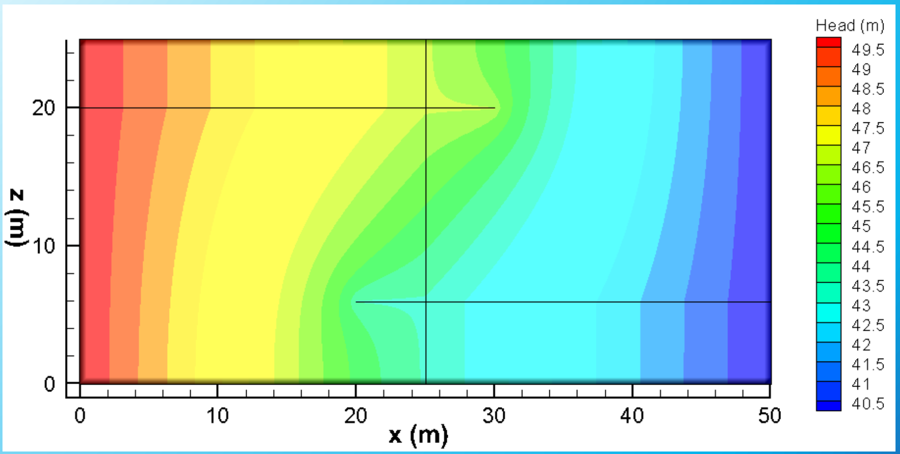
Conceptual Model



Things to try:

- Change fracture aperture
- Move fracture locations
- Add fractures

Sample Simulation Results



Module 4b

Fully Saturated	Homogeneous	Steady Flow, Transient Transport	Discrete Fractures	

Model Details

Node Count = 10,302

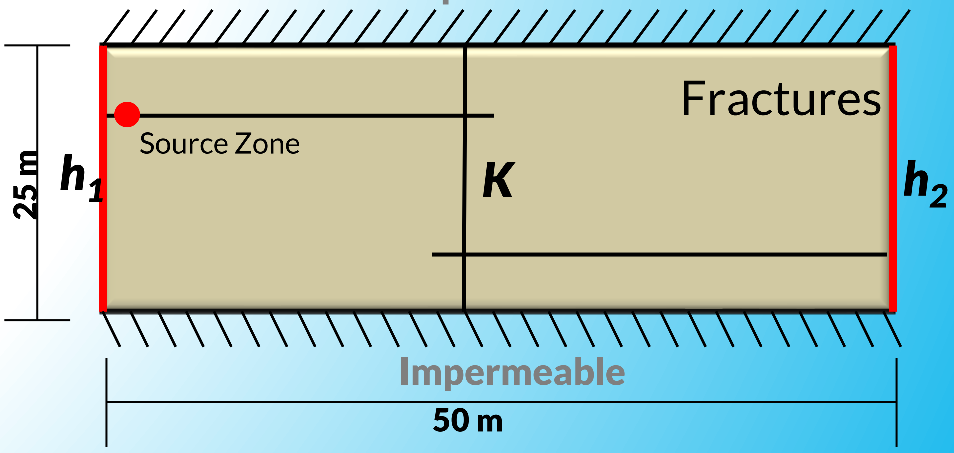
K = 1 x 10^{-9} m/s

$h1$ = 50 m

$h2$ = 40 m

Fracture Aperture = 1 x 10^{-4} m

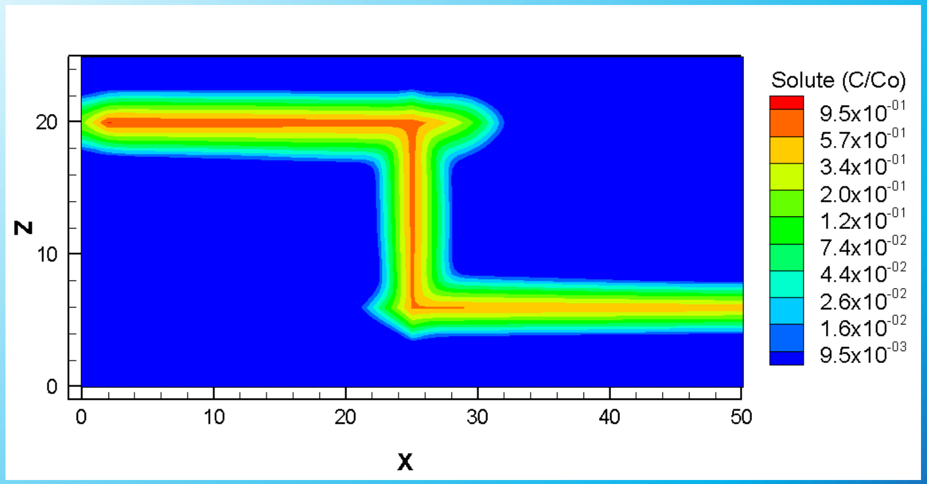
Conceptual Model



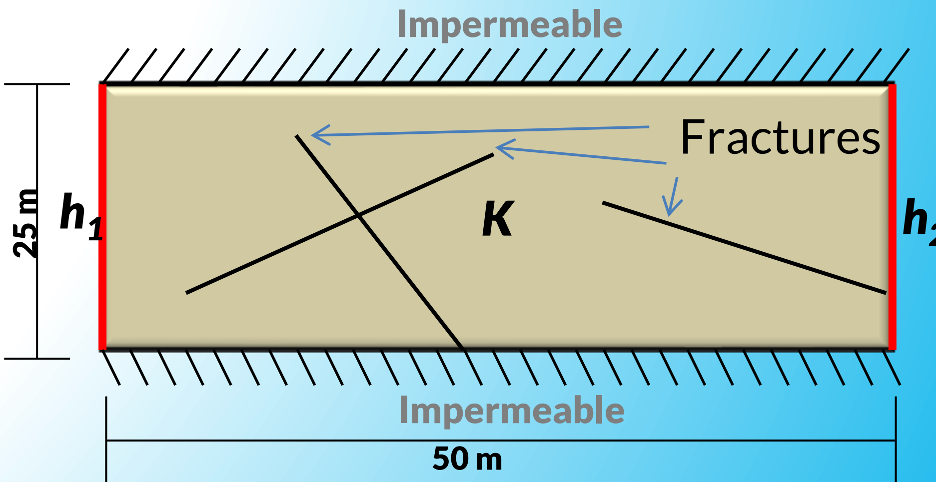
Things to try:

- Change source location
- Change fracture aperture
- Move fracture locations
- Add fractures

Sample Simulation Results



Conceptual Model



Model Details

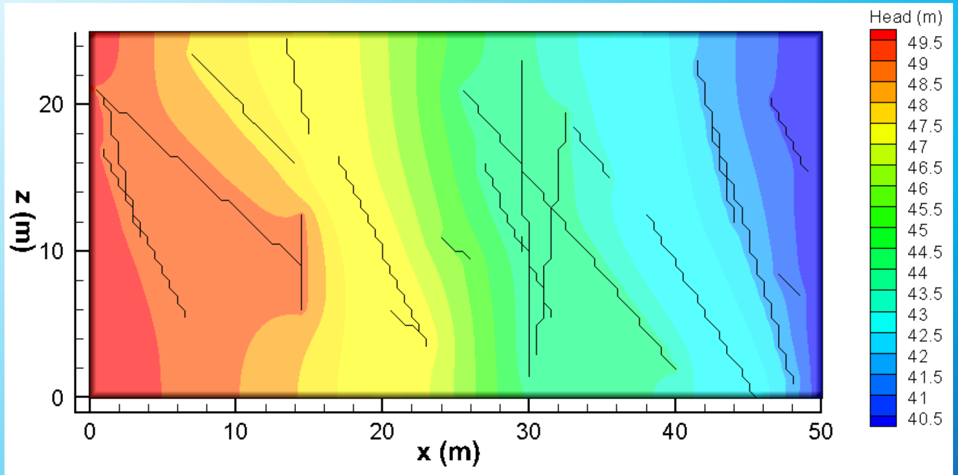
Node Count = 10,302

$K = 1 \times 10^{-9}$ m/s

h1 = 50 m

h2 = 40 m

Sample Simulation Results



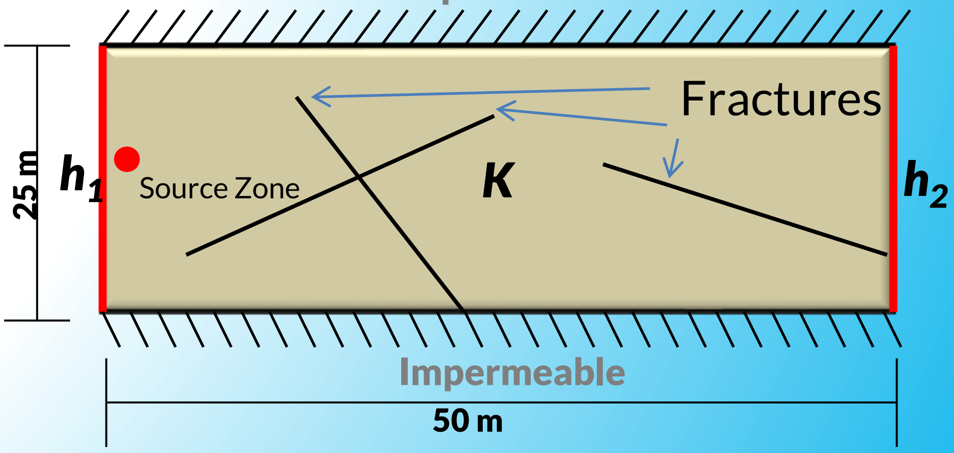
Things to try:

- Change fracture orientation distribution
- Change fracture aperture distribution
- Change fracture length distribution

Module 4d

Fully Saturated	Homogeneous	Steady Flow, Transient Transport	Discrete Fractures (Random)	

Conceptual Model



Model Details

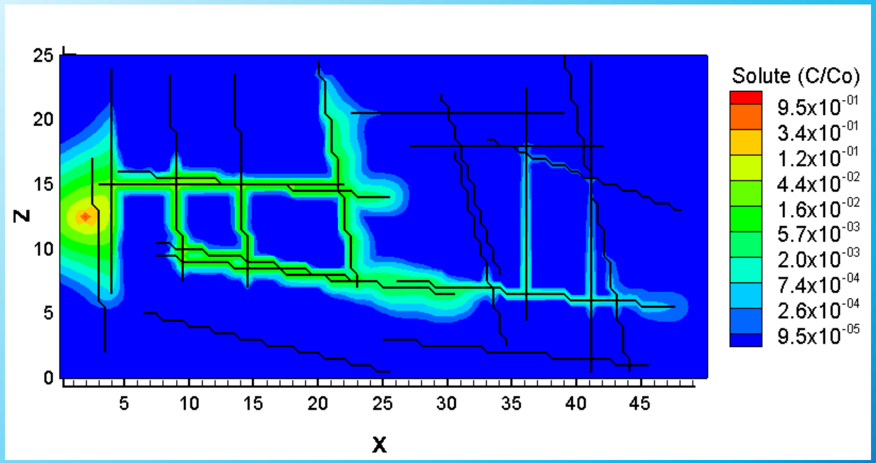
Node Count = 10,302

$K = 1 \times 10^{-6}$ m/s

h1 = 50 m

h2 = 40 m

Sample Simulation Results



Things to try:

- Change source location
- Change fracture orientation distribution
- Change fracture aperture distribution
- Change fracture length distribution
- Change seed for random generation