Tutorial 3: Layering

The purpose of this tutorial is to demonstrate the use of layering to structure your project. Layers are part of the DSM2 data management system. They allow input items to be grouped in logical bundles, and allow changes to be brought into an old simulation without erasing or altering archived items. At the same time we will neaten up our input by dividing it into several files that are "included" from a fairly sparse primary file.

1. Convert the previous hydro.inp GRID items to external files

In order to use layers, the input tables have to be gathered into individual input files. Each file represents a layer, and files that are read later replace files that are read earlier. Parent and child tables (e.g. channel and xsect) must be grouped in the same files.

a. Move the channel and reservoir data:
   1) Navigate to the tutorial directory.
   2) Create a new file in Notepad++ called grid_tutorial_base.inp
   3) Open hydro.inp.
   4) Locate the CHANNEL and XSECT_LAYER blocks. Cut them and paste them into grid_tutorial_base.inp
   5) Locate the RESERVOIR and RESERVOIR_CONNECTION blocks. Cut them and paste them into grid_tutorial_base.inp and save the file.
   6) Similarly move the TRANSFER and GATE information.
   7) Make sure the data have been removed from hydro.inp.
   8) Now add these lines that will tell DSM2 you want to include data from other files and that these files will contain GRID (channel, reservoir, transfer and gate) blocks and their child tables.

```
GRID
grid_tutorial_base.inp
END
```

2. Creating and Disabling a new reservoir:

In this section, we will learn how to remove a feature by adding a new layer and disabling the feature. We don't want to mess too much with what we have already, so we are going to add a dummy reservoir in our grid_tutorial_base layer. A second file will then be added to mask and delete it.

1) Create a new Reservoir in *tutorial_grid_base*
    i) In tutorial_grid_base.inp, enter data for the new reservoir below the data for res_1
    ii) Name: *dummy_res*
    iii) Area (million sq ft): *60*
    iv) Bottom elev (ft): -30

b. In the *Reservoir Connections* table:
    1) Right-click and select *Insert row*.
    2) Enter the following values into the appropriate fields:
        i) Reservoir name: dummy_res
        ii) Node: *5*
        iii) Res Coef (in): *220*
        iv) Res Coef (out): *220*

c. Again, in the *Reservoir Connections* table:
    1) Enter the following values into the appropriate fields:
        i) Reservoir name: dummy_res
        ii) Node: *6*
        iii) Res Coef (in): *220*
        iv) Res Coef (out): *220*

d. Save the current settings.

3. Create a Reservoir Revision Layer:

1) Create a file called *grid_tutorial_revision.inp.* Add this file to your GRID include-file section, which will now look like this:

```
GRID
grid_tutorial_base.inp
grid_tutorial_revision.inp
END
```

The include files will be prioritized in order they are read, later files replacing earlier ones. When a parent table identifier (usually a channel/node number or a “name”) is overridden by a later file, its data (including child tables) will be ignored. Everything will come from the higher layer.

2) Copy the reservoir table header and data from tutorial_grid to *grid_tutorial_revision.inp*.
3) Prepend a carat too the item in the parent table. (^). Your entry should look like this:

```
RESERVOIR
NAME         AREA  BOT_ELEV
^dummy_res   60.0     -30.0
END

##
RESERVOIR_CONNECTION
RES_NAME     NODE    COEF_IN         COEF_OUT
dummy_res       5      220.0            220.0
END
```

b. By overriding the name “dummy_res” and also marking it unused, you have now effectively removed dummy_res from the calculations. The child table is automatically ignored as well (so in a sense the entries there are unnecessary). How is this different from commenting out “dummy_res”?
c. Save the current settings.

4. Altering the Properties of the Original Reservoir res_1:

a. In the *Reservoirs* table of grid_tutorial_revision.inp, change the *Area (million sq ft)* field of res_1 from *40* to *50*.
b. Copy the RESERVOIR_CONNECTION entries for res_1 from grid_tutorial_base to grid_tutorial_revision. Why is this necessary? When you override a layer (file) with another entry in a parent table that has the same identifier, you COMPLETELY replace that item in the new layer including child items.
c. Save the current settings.

5. Changing the name of Channel 2004:

In this step, we will replace the channel number of Channel 2004. In this case, what we are changing is the identifier itself, rather than the parameters and data. So what we will do is delete Channel 2004 and put in a Channel 4 that is identical. In the process, we will ignore this change in other parts of the input and see what happens to initial conditions and output requests that reference a non-existent channel.

a. Keep the *grid_tutorial_revision* file open.
b. Navigate to tutorial_grid and copy the channel and xsect data from tutorial_grid to grid_tutorial_revision. Keep only channel 2004.
c. In the *Channels* and XSECT tables, copy the data for Channel 2004.
d. In one of your two copies of channel 2004, change the channel number in both tables to 4.
e. Eliminate channel 2004 by prepending a carat. Your revision should look like this:

```
CHANNEL
CHAN_NO LENGTH MANNING DISPERSION UPNODE DOWNNODE
^2004    15000   0.035        0.3      4        5
4        15000   0.035        0.3      4        5
END



XSECT_LAYER
CHAN_NO DIST  ELEV   AREA WIDTH WET_PERIM
2004     0.5 -24.0    0.0  40.0      40.0
2004     0.5   0.0  960.0  80.0     91.22
2004     0.5  20.0 2640.0 160.0     133.6
4        0.5 -24.0    0.0  40.0      40.0
4        0.5   0.0  960.0  80.0     91.22
4        0.5  20.0 2640.0 160.0     133.6
END
```

f. Save your work. Note that the entries in XSECT_LAYER for channel 2004 are redundant. You will often include complete entries as you toggle things “on” or “off” – otherwise feel free to delete the rows.

6. Add Initial Conditions for the New *Channel 4*:

Since there is no default initial condition for channel 4, we will have to add one.

a. Create a file called channel_ic_revision.inp.
b. Copy the CHANNEL_IC table headers from channel_ic_tutorial.inp to the new file.
c. Create two rows of data for channel 4:

```
CHANNEL_IC
CHAN_NO   DISTANCE   STAGE       FLOW
4                0     0.0        0.0
4           length     0.0        0.0
END
```

d. Now every channel has an initial condition. Do you need to do something about Channel 2004? Try and see.

7. Running HYDRO and QUAL

a. In Windows Explorer, navigate to the directory: *{DSM2_home}**tutorial**simple*\.
b. Right-click on the directory, *t3_layering*, and select *Open Command Window Here*.
c. In the command window, type: *hydro hydro.inp*.
d. In the command window, type: *qual qual.inp*.
e. Open the *output.dss* file in the *t3_layering* directory, and examine the results.
f. Open echo.inp. This is an “echoed input” that replicates your input verbatim, except ENVVAR replacements have been made and all the channel xsects are in the one-file format. You should be able to run the model using this file as easily as with the original hydro.inp. Take a look and see:
    1) Did channel 4 get in the input?
    2) Did channel 2004? What does this mean?

8. Converting hydro.inp to input blocks

Now let’s convert hydro.inp completely to include files except for the SCALAR and IO_FILE sections. In future tutorials, hydro and qual simulations will be organized this way. The file hydro.inp or qual.inp is usually reserved for scalar or input/output file designations.

a. Create an INITIAL_CONDITION include block underneath the GRID include block:

```
INITIAL_CONDITION
ic_tutorial.inp
END
```

b. Create a file called ic_tutorial.inp
c. Cut (not copy) the CHANNEL_IC and RESERVOIR_IC data and paste it into this file.
d. Create an include block called HYDRO_TIME_SERIES.

```
HYDRO_TIME_SERIES
input_boundary_hydro_tutorial.inp
input_transfer_flow_tutorial.inp
END
```

e. Create a file called boundary_hydro_tutorial.inp. Cut (not copy) the BOUNDARY_STAGE and BOUNDARY_FLOW input from hydro.inp to boundary_hydro_tutorial.inp.
f. Similarly, create a file called transfer_flow_tutorial.inp. Cut and paste the TRANSFER_FLOW data into this file.
g. Create an include block called OUTPUT_TIME_SERIES.

```
OUTPUT_TIME_SERIES
output_hydro_tutorial.inp
END
```

h. Similarly, create the file called output_hydro_tutorial.inp. Cut and paste the OUTPUT_CHANNEL data into this file.
i. The remaining tutorials will use include blocks extensively for both hydro and qual.

9. Running HYDRO and QUAL with all include files

a. In the SCALAR section of hydro.inp, change DSM2MODIFIER to include_block
b. In Windows Explorer, navigate to the directory: *{DSM2_home}**tutorial**simple*\.
c. Right-click on the directory, *t3_layering*, and select *Open Command Window Here*.

d. In the command window, type: *hydro hydro.inp*.

e. In the command window, type: *qual qual.inp*.

f. Open the *output.dss* file in the *t3_layering* directory, and examine the results, comparing it to the last run. Did putting things in input blocks change anything?

10. Learning more

Overriding is easy to understand. The main things you will need to keep in mind are

a) Understanding how child table replacement works:
   a. You can't replace the child element without replacing the parent.
   b. The children of an overridden parent element are never used.

b) What the identifier is for each row in a table. In most cases this is the first field and it is usually a name or a map number (it is a label rather than a piece of hard data). In some cases (e.g. output), the unique identifier may be two fields such as NAME and VARIABLE for output. Overriding only occurs when the identifier for the row is duplicated. This information is available in the table reference documentation in the folder /dsm2/doc.

c) Which data can be included in which blocks. For instance, GRID can contain CHANNEL, GATE, RESERVOIR and TRANSFER data. This information is given below:

Include Block	Sections
CONFIGURATION	ENVVAR SCALAR
GRID	CHANNEL XSECT (child) XSECT_LAYER (child) RESERVOIR RESERVOIR_CONNECTION (child) GATE GATE_DEVICE (child) TRANSFER

GROUPS	GROUP GROUP_MEMBER (child)
HYDRO_TIME_SERIES	INPUT_TRANSFER_FLOW INPUT_GATE BOUNDARY_STAGE BOUNDARY_FLOW SOURCE_FLOW SOURCE_FLOW_RESERVOIR
INITIAL_CONDITION	CHANNEL_IC RESERVOIR_IC
OPERATION	OPERATING_RULE OPRULE_EXPRESSION OPRULE_TIME_SERIES
OUTPUT_TIME_SERIES	OUTPUT_CHANNEL OUTPUT_RESERVOIR OUTPUT_CHANNEL_SOURCE_TRACK OUTPUT_RESERVOIR_SOURCE_TRACK OUTPUT_GATE
PARTICLE	PARTICLE_INSERTION PARTICLE_FLUX_OUTPUT PARTICLE_GROUP_OUTPUT
QUAL_SPATIAL	RATE_COEFFICIENT
QUAL_TIME_SERIES	INPUT_CLIMATE NODE_CONCENTRATION RESERVOIR_CONCENTRATION