## Tutorial for GP-SWAT

GP-SWAT is a two-layer model parallelization tool to reduce the run time of SWAT model through a combination of model spatial decomposition and the graph-parallel Pregel algorithm. GP-SWAT consists of two components: a preprocessing program and a driver program. The preprocessing program is used to extract route information from the watershed configuration file (fig.fig) of the SWAT model, and create a watershed configuration file for each subbasin model. The second component, the driver program, creates a hydrological model property graph from the route information generated by the preprocessing program and defines how the models are to be simulated in parallel; its output is then replicated to the executors to actually carry out model parallelization. There are four steps that are involved for achieving two-layer model parallelization.

## Step one: setup running environment

GP-SWAT runs on top of Spark, which can run in a standalone or cluster mode. For information on how to setup Spark on cluster mode, please refer to Spark document (<https://spark.apache.org/docs/latest/>); For information on how to launch Spark on standalone mode on a Windows OS, please refer to the tutorial by Ravichandra (<https://www.knowledgehut.com/blog/big-data/how-to-install-apache-spark-on-windows>).

## Step two: extract route information

* Copy the watershed model to directory “modelbackup” under current directory
* Copy fig.fig file to the current directory from “modelbackup” directory
* Run the “extract route information.bat” batch process file
* Three file (subbasin.txt, stream.txt, routeset.txt) should be generated if extract route information.bat was successfully executed

Input for this step:

* Model input files in the “modelbackup” directory
* Watershed configuration file “fig.fig”
* Jar files required to perform this step, including graphswat.jar and commons-io-2.0.jar
* Batch process file “extract route information.bat”

Output of this step:

* Stream.txt
* Subbasin.txt
* Routeset.txt

Please also refer to step2.wmv for information on how this step is executed.

## Step three: create subbasin configuration files

* Make sure step 2 was successfully executed
* Make a directory named as “wconfigfiles” at current directory
* Run the “create config files.bat” batch process file
* Configuration files for each subbasin and simulation should be generated if “create config files.bat” was successfully executed

Input for this step:

* Route information file “routeset.txt”
* Watershed configuration file “fig.fig”
* Jar files required to perform this step, including graphswat.jar and commons-io-2.0.jar
* Batch process file “create config files.bat”

Output of this step:

* Configuration files for each subbasin and simulation

Please also refer to step3.wmv for information on how this step is executed.

## Step four: run GP-SWAT

* Make sure step 2 and step 3 were successfully executed
* Replicate model in the models directory
* Run the “run gpswat.bat” batch process file
* Simulation results for each subbasin and simulation should be generated at the “repository” if “run gpswat.bat” was successfully executed

Input for this step:

* Model replications under the “models” directory
* Driver program “driver.scala” under the “scala” directory
* Jar files required to perform this step, including graphswat.jar and commons-io-2.0.jar
* Batch process file “run gpswat.bat”

Output of this step:

* Simulation results for each subbasin and simulation

Please also refer to step4.wmv for information on how this step is executed.