**HRU- delin : délimitation des HRU**

**version batch UNIX/LINUX**

version de HRU-delin : v4

date de création : 7-12-2016

Contenu

[1 Présentation 1](#_Toc454463212)

[2 Implémentation 2](#_Toc454463213)

[3 Installation 2](#_Toc454463214)

[4 Organisation 2](#_Toc454463215)

[5 Déroulement du processus 3](#_Toc454463216)

[6 Paramétrage des étapes 3](#_Toc454463217)

[6.1 Codification et règles de gestion des fichiers 4](#_Toc454463218)

[6.2 Etape 1 : préparation des couches 4](#_Toc454463219)

[6.2.1 Discrétisation des altitudes, pentes et orientations 4](#_Toc454463220)

[6.3 Etape 2 : flux orientés et bassins aux stations 5](#_Toc454463221)

[6.3.1 Relocalisation des stations 5](#_Toc454463222)

[6.4 Etape 3 : construction des HRUs 5](#_Toc454463223)

[6.5 Etape 4 : topologie et génération des paramètres pour le modèle 6](#_Toc454463224)

[7 Lancement de HRU-DELIN 6](#_Toc454463225)

[8 Reprise entre les étapes 6](#_Toc454463226)

[9 Résultats 7](#_Toc454463227)

[9.1 Calcul des caractéristiques des HRU 7](#_Toc454463228)

[10 Quelques messages d’erreur 7](#_Toc454463229)

[10.1 Step2 7](#_Toc454463230)

[Annexe 1 : Etapes du workflow HRU-DELIN\_BATCH 9](#_Toc454463231)

[Annexe 2 : Composants HRU-delin 10](#_Toc454463232)

[Annexe 3 : Fichier de configuration 12](#_Toc454463233)

[Annexe 4 : exemple de fichier de configuration 15](#_Toc454463234)

[Annexe 5 : lancement de HRU-delin sous Pénélope 18](#_Toc454463235)

[1. Pénélope, connexion sur Pénélope 18](#_Toc454463236)

[2. HRU-delin sous Pénélope 18](#_Toc454463237)

[3. Données pour HRU-delin sous Pénélope 18](#_Toc454463238)

# Présentation

HRU-delin (pour HRU delineation) est un outil de construction de maillage support d’une modélisation hydrologique distribuée. Les unités du maillage sont les HRU (Hydrological Response Unit). Un HRU est une zone qui présente une réponse homogène du point de vue hydrologique.

HRU-delin construit le maillage à partir d’un Modèle Numérique de Terrain, de données caractérisant le sol sous forme de rasters et d’un shapefile des stations de jaugeage.

Il génère des fichiers paramètres destinés à la modélisation sous J2000.

A partir du MNT sont dérivés :

- altitude, pente, orientation

- flux orientés : directions de drainage, accumulation, sous-bassins topographiques, cours d’eau, sous-bassins des stations de jaugeage

Les données additionnelles sont en général le type de sol, l’occupation du sol, la géologie. Elles sont optionnelles.

La délimitation des HRU résulte de la combinaison (overlay) des couches :

- sous-bassins topographiques et sous-bassins aux stations

- données additionnelles

- altitude, pente et orientation discrétisées

En raison de la variabilité spatiale des caractéristiques, on retrouve en sortie de l’overlay de très nombreux pixels isolés qui représentent, chacun, une combinaison unique. Ils vont être regroupés et/ou rattachés à un HRU. En fonction de l’étendue de la zone étudiée et de sa résolution cette opération peut être très longue. À l’issue de cette étape on obtient le maillage.

Pour permettre une modélisation distribuée il faut que le maillage ait une topologie (ie : connexions entre les mailles ou HRU). Les HRU et les cours d’eau sont analysés pour déterminer leurs relations : vers quelle entité un HRU se déverse-t-il ? Cette étape complexe peut également prendre du temps.

À la fin du traitement on obtient un ensemble de fichiers composé de :

- ***hru.par*** paramètres des HRU

- ***reach.par*** paramètres des cours d’eau

- ***hrus.shp*** shapefile des HRU créés

- ***streams.shp*** shapefile des cours ‘eau

La version présentée dans ce document est un ensemble de programmes à installer sur une machine UNIX/LINUX et s’exécute en ligne de commande.

HRU-delin a été développé à partir du logiciel GRASS-HRU de l’université de JENA. Il est écrit en PYTHON et utilise les modules de traitement géographique du SIG GRASS.

# Implémentation

La délimitation des HRU en batch s’effectue en 4 étapes (schéma en [annexe 1](#_Annexe_1_:)) :

1. préparation des couches pour les étapes suivantes, dérivation des flux et des sous-bassins

2. relocalisation des stations sur les cours d’eau dérivés du MNT par calcul, découpage des cours d’eau aux stations, dérivation des bassins aux stations

3. délimitation des HRU

4. calcul de la topologie et création des fichiers paramètres (***hru.par*** et ***reach.par***) destinés à la modélisation

# Installation

\*\* L’interpréteur de commande UNIX est sensible à la casse ⭢ respecter les majuscules/minuscules

Le package a télécharger est sur la forge IRSTEA

* Télécharger ***HRU-DELIN\_batch\_unix\_install\_numversion.zip*** qui contient les composants (modules, fichiers, …) et leur script d’installation
* Décompresser l’archive
* Ouvrir un terminal
* Dans le répertoire où l’archive a été décompressée, se placer dans **HRU-delin\_batch\_unix\_install\_**
* Rendre exécutable le script d’installation : **chmod 755** **./hru-delin\_install.sh**
* Taper la commande : **sudo ./hru-delin\_install.sh** (rq : la commande ***sudo***, qui permet d’exécuter une commande avec les privilèges de l’administrateur, requiert le mot de passe de l’utilisateur)
* L’utilisateur peut spécifier un répertoire sinon l’installation se fera sous ***home/user/hru-delin***

Packages installés :

grass : Geographic Ressources Analysis System Support GIS

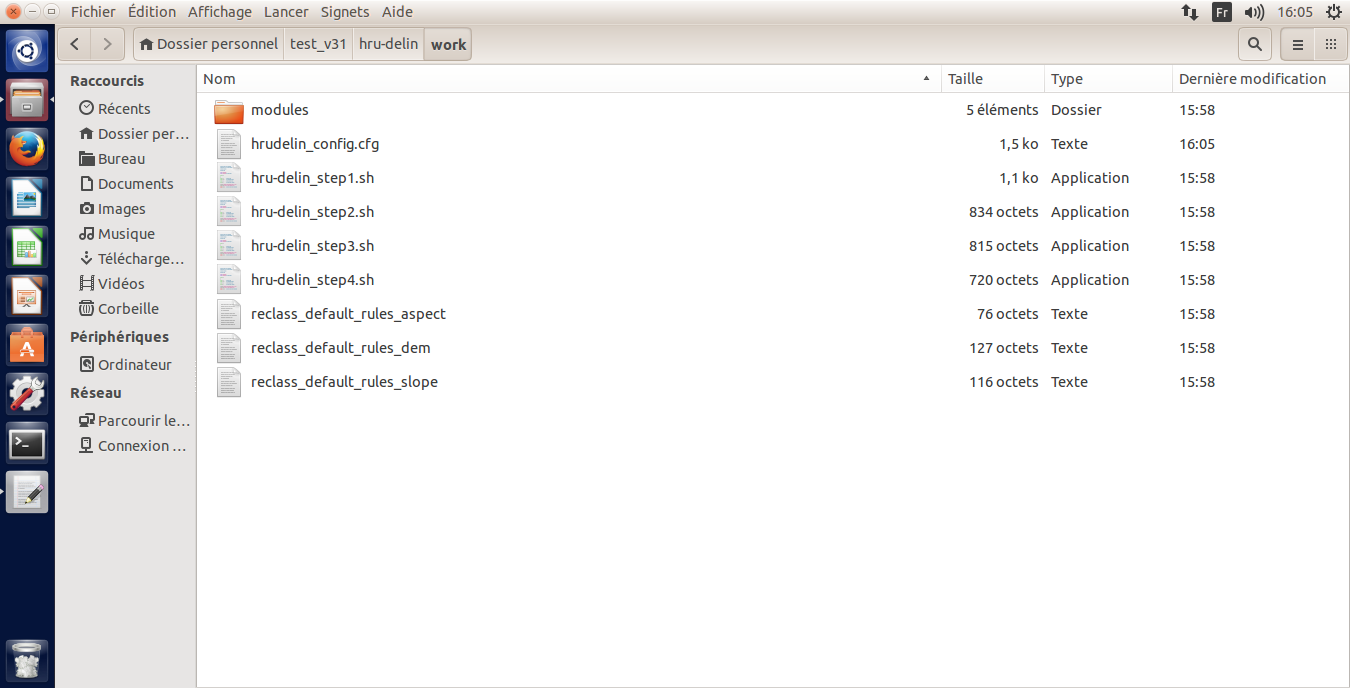
python-gdal : Liens avec Geospatial Data Abstraction Library qui fournit des classes d’objet pour manipuler rasters (GDAL) et vecteurs (OGR)

default-jre Environnement d’exécution java (pour le module ***zirkelkill***)

# Organisation

Les composants de HRU-delin fournis à l’installation sont listés en [annexe 2](#_Annexe_2_:).

Le répertoire d’exécution est ***hru-delin/work.***



Les sources des modules PYTHON et awk sont dans le répertoire ***modules.*** Les exécutables java (***zirkelkill*** : module de détection de cycle dans la topologie) sont dans ***/usr/share/java***.

Le module **r.reach.par** dans ***/usr/lib/grass64/scripts*** (répertoire standard des modules GRASS).

***tmp*** est un répertoire de travail pour les différentes phases du processus.

Dans le fichier de configuration (***hrudelin\_config.cfg***, voir [Paramétrage des étapes](#_Paramétrage_des_étapes_1)), l’utilisateur indique où se trouvent les fichiers en entrée (MNT, stations, données additionnelles) et où il souhaite stocker les fichiers en sortie (fichiers inter-étapes, résultats).

# Déroulement du processus

Il y a un script shell par étape, qui initialise les variables d’environnement GRASS et lance le module PYTHON correspondant.

1 étape = 1 script shell = 1 module

|  |  |  |
| --- | --- | --- |
|  | Nom du script shell | Nom du module PYTHON |
| 1 | hru-delin\_step1.sh | hru-delin\_1\_init |
| 2 | hru-delin\_step2.sh | hru-delin\_2\_basins |
| 3 | hru-delin\_step3.sh | hru-delin\_3\_hrugen |
| 4 | hru-delin\_step4.sh | hru-delin\_parms\_model |

Tableau 1: Noms des scripts shell et modules PYTHON

En particulier ***hru-delin\_step1.sh*** crée les répertoires de la base de données GRASS (répertoire ***grass\_db***) : **LOCATION** et **MAPSET** et les fichiers **DEFAULT\_WIND** et **WIND** .

# Paramétrage des étapes

Le fichier ***hrudelin\_config.cfg*** (dans ***hru-delin/work***) contient les paramètres à passer aux modules de traitement : répertoires des fichiers en entrée et en sortie, emprise, taille min des bassins, … . Il est détaillé en [Annexe 3 : Fichier de paramétrage](#_Annexe_3_:_1) et un exemple sur le bassin du Rhone est donné en [Annexe 4 : exemple de fichier de configuration](#_Annexe_4_:).

Afin de conserver la trace du paramétrage d’un passage, ce fichier est sauvegardé à la dernière étape dans le répertoire des fichiers intermédiaires, sous le nom ***hrudelin\_config.cfg***.

## Codification et règles de gestion des fichiers

1. Les noms de fichier doivent spécifier l’extension. Exemples : MNT\_Rhone**.tif** stations**.shp**
2. Le tilde (~) dans les chemins n’est pas accepté.
3. Les fichiers créés par HRU-DELIN sont déposés dans les répertoires :
   * **[dir\_out] files** : **répertoire des fichiers intermédiaires** : couches de travail, paramètres de classifications, …
   * **[dir\_out] results**: **répertoire des résultats** délivrés en fin de traitement :
     1. *hru.par* : paramètres pour la modélisation
     2. *reach.par*

**Ces répertoires doivent être distincts**.

Exemple :

[dir\_out] files : *BVtest\_intermediaires*

[dir\_out] results: *BVtest\_resultats*

## Etape 1 : préparation des couches

L’emprise de la zone de travail est définie par l’utilisateur sous l’une des formes :

- par défaut : étendue totale du MNT

- par un polygone

- par des coordonnées géographiques

Toutes les couches seront découpées selon ce gabarit.

A cause des limitations d’un module de manipulation des objets vecteur (bibliothèque OGR), dans le cas d’un shapefile des stations ne comprenant qu’une seule entité il faut fournir un polygone.

### Discrétisation des altitudes, pentes et orientations

Les valeurs d’altitude, pente et orientation sont regroupées en classes. Les règles de discrétisation sont stockées dans des fichiers texte.

Une version de base est proposée dans les fichiers ***reclass\_default\_rules\_dem***, ***reclass\_default\_rules\_slope*** et ***reclass\_default\_rules\_aspect***.

C’est le paramètre [reclass\_] du fichier de configuration qui pilote le traitement :

* rules\_auto\_ = yes : les règles de recodification seront construites par le module **hru-delin\_1\_init**, avec uniquement pour le MNT le paramètre spécifique ***step\_dem*** pour indiquer la valeur de l’intervalle
* rules\_auto \_ = no : l’utilisateur fournit le fichier des règles :
  + copie du fichier ***reclass\_default\_rules\_*** en ***reclass\_rules\_***

 ne pas modifier directement les fichiers default !!! En cas de fausse manip, restaurer le fichier modifié à partir de HRU-DELIN\_BATCH\_install (voir la liste des composants en annexe 2)

* + codification des règles :

borne inf:borne sup:classe:classe

exple : 10.0:20.0:20:20

les pentes dans l’intervalle [10.0 ;20.0[ seront recodées à 20

Ce choix implique que l’on connait les valeurs min et max du raster.



S’il n’y a pas de règle pour une valeur, le pixel sera fixé à ‘no data’.

## Etape 2 : flux orientés et bassins aux stations

### Relocalisation des stations

A l’étape 1 les flux orientés et les sous-bassins ont été dérivés du MNT. Les stations sélectionnées ne se retrouvent probablement pas sur les cours d’eau calculés. Pour générer les bassins-versants aux stations, certaines devront être replacées sur les cours d’eau. L’utilisateur peut soit fournir un nouveau fichier des stations, soit demander que cette relocalisation des stations soit faite par le module.

#### Fichier des stations relocalisées

Le fichier est indiqué dans le paramètre **[relocated] new\_gauges**. Il doit se trouver dans le répertoire **[dir\_in]**.

#### Relocalisation automatique des stations

A priori la distance des stations aux cours d’eau dérivés topographiquement doit être minime. Mais il est aussi possible que le shape de description des stations comporte des incertitudes quant à la position réelle des stations aussi bien que sur la surface drainée.

Le déplacement d’une station se fait en fonction de critères de tolérance fournis par l’utilisateur et qui portent sur la surface drainée et la distance de la station actuelle, et peut s’effectuer en 2 phases.

Les stations dont la surface drainée est inférieure à la surface de BV min (paramètre ***basin\_min\_size***) sont supprimées.

Principe : Recherche dans un périmètre max (paramètre ***distance\_tolerance***) du pixel le plus proche de la position initiale de la station et dont la valeur est comprise dans l’intervalle surface drainée ± ***surface\_tolerance***.

Exemple surface drainée de la station = 52 km²

surface\_tolerance\_1 = 10

distance\_tolerance\_1 = 10

⭢ recherche 10 pixels autour de la position initiale de la station du pixel dont la surface drainée est la plus proche de celle de la station, dans un écart compris entre 46,8 et 57,2 km²

Si une première recherche (avec les paramètres ***surface\_tolerance\_1*** et ***distance\_tolerance\_1***) n’a pas permis de relocaliser la station, une seconde sera entreprise avec ***surface\_tolerance\_2*** et ***distance\_tolerance\_2*** si ceux-ci sont renseignés.

Les déplacements de station effectués sont enregistrés dans un fichier log (***auto\_reloc.csv***) dans le répertoire des fichiers intermédiaires.

Pour celles qui n’ont pas pu être déplacées, un traitement manuel sera nécessaire. Il suffira alors de relancer l’étape 2 avec le nouveau shapefile des stations (voir [Fichier des stations relocalisées](#_Nouveau_fichier_des))

## Etape 3 : construction des HRUs

Dans cette étape les couches sont combinées pour créer les HRUs :

- les couches subbasins et watersheds : obligatoires

- les données additionnelles qui ont été spécifiées : géologie, sols, …

- les dérivées du MNT – altitude, pente, orientation – sont facultatives et peuvent entrer ou

non dans l’overlay, au choix de l’utilisateur : section [layer\_overlay] de ***hrudelin\_config.cfg***.

La taille minimale de HRU est donnée en pixels ***[hru\_min\_surface].***

L’analyse s’effectue par sous-bassin topographique. Si la surface d’un sous-bassin est inférieure à **[*hru\_min\_surface*]** il est d’office converti en HRU.

Ceci a 2 conséquences :

- un HRU peut avoir une surface inférieure à ***hru\_min\_surface***

- il est possible de forcer le maillage au découpage en sous-bassins topographiques en spécifiant un ***hru\_min\_surface*** = surface du plus grand sous-bassin.

## Etape 4 : topologie et génération des paramètres pour le modèle

A cette étape la topologie du maillage est construite, on la retrouve dans les colonnes ***to\_poly*** et ***to\_reach*** du fichier paramètres pour la modélisation.

Le module ***zirkelkill*** (développé à l’université de JENA) analyse cette topologie et lorsqu’un cycle est détecté la connexion d’un des HRU du cycle est forcé avec l’identifiant de sous-bassin topographique. Ceci explique qu’on peut parfois retrouver un HRU (colonne ***to\_reach***) se déversant dans un cours d’eau alors qu’il n’y est géographiquement pas relié.

# Lancement de HRU-DELIN

Commande de lancement d’un script : se placer dans le répertoire ***hru-delin/work***

**./hru-delin\_step%.sh** où % = 1,2,3 ou 4

# Reprise entre les étapes

Il est possible de relancer une étape, sans repartir du début.

A la fin de chaque étape les fichiers nécessaires à la suite du processus sont stockés dans le répertoire spécifié dans le paramètre **[dir\_out] files** : du fichier ***hrudelin\_config.cfg***.

A noter :

* ce répertoire est recréé par la première étape. Pour conserver les traces d’un run il faut le sauvegarder ou le renommer
* les règles de recodification utilisées y sont également sauvegardées

Le nom des fichiers est construit de la manière suivante :

step%\_origine+type.extension

% = 1,2 ou 3

origine : nom du MNT d’origine

type : ce qu’il représente : pente, bassin d’accumulation, …..

extension : tif pour l’instant

exemple : step1\_dgm\_aspect\_reclass.tif

fichier raster créé en step1, orientations dérivées du MNT ‘dgm’ classifiées

# Résultats

A la fin de l’étape 4, les fichiers pour la modélisation sont délivrés dans le répertoire spécifié par **[dir\_out] results** du fichier ***hrudelin\_parms.cf****.*

shapefile des HRU : ***hrus.shp***

paramètres pour modélisation : ***hru.par*** et ***reach.par***

## Calcul des caractéristiques des HRU

Les données caractéristiques des HRU sont mises en forme dans le fichier de paramètres pour la modélisation.

Pour chaque HRU on retrouve les colonnes :

|  |  |
| --- | --- |
| area | résolution horizontale x résolution verticale x nb pixels |
| elevation | altitude moyenne |
| slope | pente moyenne en degré |
| aspect | moyenne des orientations les plus proches du mode de la série des orientations reclassées |
| centroid | coordonnées (x,y) du centroïde du polygone HRU |
| watershed | identifiant du sous-bassin à la station, désigné par le paramètre [col\_name](#col_name) |
| subbasin | identifiant du sous-bassin topographique |
| ID1 | données additionnelles |
| ID2 | données additionnelles |
| ID3 | données additionnelles |
| to\_poly | si ≠ 0 c’est l’id de la HRU suivante dans la topologie  si = 0 la HRU se déverse dans un brin (colonne to\_reach) |
| to\_reach | si ≠ 0 c’est l’identifiant du cours d’eau |
| irrig | 0 : HRU non irrigué  1 : HRU irrigué |

# Quelques messages d’erreur

**ERROR 6: SetColorTable() only supported for Byte or UInt16 bands in TIFF format.**

Ce message est émis par GDAL au moment de l’export d’un raster en raison d’une limitation sur les GeoTiff. C’est en fait un ‘warning’. Les données sont correctement exportées mais pas la ColorTable.

**min\_size = int(parms.get('basin\_min\_size', 'size'))**

**ValueError: invalid literal for int() with base 10: ''**

Le paramètre doit être numérique (ici **'*[basin\_min\_size] size*'**)

## Step2

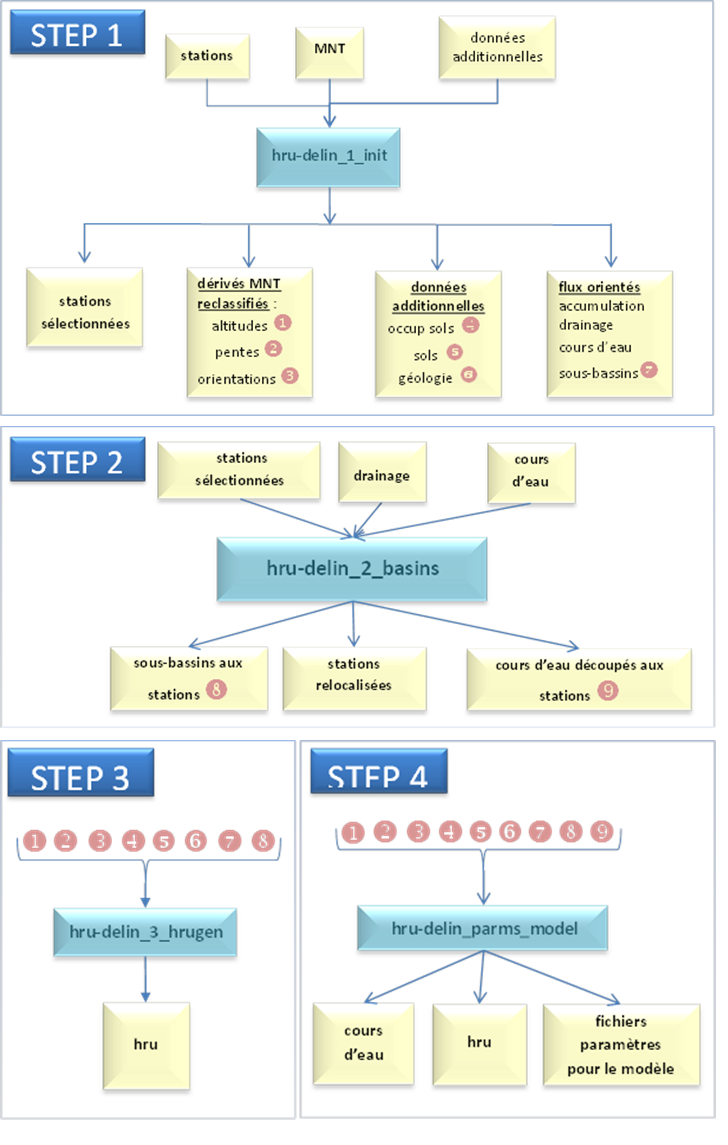
**Column 'zorglub' not found**

Error in db\_open\_select\_cursor()

ERROR: Cannot open cursor: 'SELECT zorglub FROM gauges\_reloc WHERE cat = 1'

Le nom de colonne spécifié dans le paramètre ***[for\_watershed\_id] col\_name*** n’existe pas.

# Annexe 1 : Etapes du workflow HRU-DELIN\_BATCH



# Annexe 2 : Composants HRU-delin

Le répertoire **HRU-DELIN\_BATCH\_install** contient les composants de base.

hru-delin\_install.sh

manuel\_utilisation.docx

hru-delin\grass\_db\location\_parms

hru-delin\grass\_db\location\_parms\_old

hru-delin\grass\_db\location\_parms\DEFAULT\_WIND

hru-delin\grass\_db\location\_parms\WIND

hru-delin\grass\_db\location\_parms\_old\DEFAULT\_WIND

hru-delin\grass\_db\location\_parms\_old\PROJ\_INFO

hru-delin\grass\_db\location\_parms\_old\PROJ\_UNITS

hru-delin\grass\_db\location\_parms\_old\WIND

hru-delin\work\hru-delin\_step1.sh

hru-delin\work\hru-delin\_step2.sh

hru-delin\work\hru-delin\_step3.sh

hru-delin\work\hru-delin\_step4.sh

hru-delin\work\hrudelin\_config.cfg

hru-delin\work\reclass\_default\_rules\_aspect

hru-delin\work\reclass\_default\_rules\_dem

hru-delin\work\reclass\_default\_rules\_slope

hru-delin\work\modules\hru-delin\_1\_init.py

hru-delin\work\modules\hru-delin\_2\_basins.py

hru-delin\work\modules\hru-delin\_3\_hrugen.py

hru-delin\work\modules\hru-delin\_parms\_model.py

hru-delin\work\modules\awk\find\_max.awk

hru-delin\work\modules\awk\find\_max\_no\_cycle\_opt.awk

hru-delin\work\modules\awk\format\_nm.awk

hru-delin\work\modules\awk\hru-delin\_1\_init.py

hru-delin\work\modules\awk\orig\_format.awk

hru-delin\work\modules\awk\repl\_header.awk

java\_modules\jdbf.jar

java\_modules\Zirkelkill2.jar

java\_modules\Zirkelkill\_1zuN.jar

reachpar\_module\r.reach.par

# Annexe 3 : Fichier de configuration

|  |  | | description |  | |
| --- | --- | --- | --- | --- | --- |
| [dir\_in] | dir | | répertoire des fichiers en entrée ( -> [files\_in] et [data]) | *chemin complet* | |
| [files\_in] | dem | | nom du MNT | *nomdufichier****.****extension* | |
| gauges | | shapefile des stations | *nomdufichier****.*shp** | |
| [data] | data | | liste des noms de fichier des données additionnelles (séparés par une virugle, sans espaces) | *geologie.tif,sols.tif,landuse.tif* | |
| [irrigation] | irrig\_rast | | raster irrigation |  | |
| [dir\_out] | files | | répertoire où seront stockés les fichiers intermédiaires | doit être différent du fichier ***results*** | |
| results | | répertoire où seront stockés les fichiers paramètres pour le modèle | doit être différent du fichier ***files*** | |
| [surface] | selection | | type de découpage du MNT | total  *surface totale du MNT*  polygon *découpage selon un shape (-> polygon)*  coords  *découpage selon des coordonnées (-> west,north, east,south)* | |
| polygon | | shapefile polygone qui servira à découper le MNT | *nomdufichier****.*shp** | |
| west  north  east  south | | coordonnées pour le découpage du MNT |  | |
| [demfill] | demfill | | Demande de traitement d’élimination des cuvettes du MNT | yes | no | |
| [reclass\_dem] | rules\_auto\_dem | | Reclassification des altitudes du MNT | yes  *exécuté par HRU-DELIN (-> step\_dem)*  no *les règles de reclassification sont spécifiées par l’utilisateur dans un fichier* ***reclass\_ rules\_dem*** | |
| step\_dem | | Valeur de l’intervalle de reclassification des altitudes | obligatoire et numérique | |
| [reclass\_slope] | rules\_auto\_slope | | Reclassification des pentes dérivées du MNT | yes  *exécuté par HRU-DELIN*  no *les règles de reclassification doivent être fournies par l’utilisateur dans un fichier* ***reclass\_rules\_slope*** | |
| [reclass\_aspect] | rules\_auto\_aspect | | Reclassification des orientations dérivées du MNT | yes  *exécuté par HRU-DELIN*  no *les règles de reclassification doivent être fournies par l’utilisateur dans un fichier* ***reclass\_rules\_aspect*** | |
| [basin\_min\_size] | size | | Taille minimale de sous-bassin (calculés par le module *r.watershed* de grass) | obligatoire et numérique  en nombre de pixels | |
| [auto-relocation] | to\_do | | Demande de relocalisation auto des stations sur les cours d’eau | yes | no | |
|  | surface\_tolerance\_1 | | écart max (pourcentage) toléré sur l’aire drainée | obligatoire et numérique si to\_do = yes | |
|  | distance\_tolerance\_1 | | éloignement max à explorer, en nombre de pixels | obligatoire et numérique si to\_do = yes | |
|  | surface\_tolerance\_2 | | écart max (pourcentage) toléré sur l’aire drainée | deuxième règle applicable (si \_1 n’a rien donné)  numérique, facultative | |
|  | distance\_tolerance\_2 | | éloignement max à explorer, en nombre de pixels | deuxième règle applicable (si \_1 n’a rien donné)  numérique, facultative | |
|  | gauge\_area\_col\_name | | nom de la colonne de la table attributaire des stations qui contient la surface drainée |  | |
|  | gauge\_area\_unit | | unité de la surface drainée | 1 : mètre  2 : km | |
|  | relocated\_gauges | | nom du shape du fichier des stations relocalisées fourni par l’utilisateur |  | |
| [for\_watershed\_id] | col\_name | | Nom de la colonne de la table du shape des stations qui servira à l’identification des bassins (calculés par r.water.outlet de grass) | *nom d’une colonne du fichier dbf associé au shapefile, dont le type doit être INTEGER* | |
| [hrus\_min\_surface] | surface | | Taille minimale de HRU (en pixel) | obligatoire et numérique. en nombre de pixels | |
| [topology] | dissolve\_cycle | | Elimination des cycles dans la topologie | y | n | |
| [layer\_overlay] | *choix des couches qui entreront dans l’overlay* | | | | |
| dem | élévation moyenne du HRU | | | x ou rien |
| slope | pente moyenne du HRU | | | x ou rien |
| aspect | orientation moyenne du HRU | | | x ou rien |

# Annexe 4 : exemple de fichier de configuration

# -----------

# environment

# -----------

[dir\_in]

dir:/home/chris/DONNEES/Rhone

[files\_in]

dem:MNT\_Rhone\_final.tif

gauges:StationsHydro\_BVRhone\_V2\_Modif\_CB\_Beaucaire.shp

[data]

hgeo:hgeo2.tif

landuse:landuse\_carre2.tif

soil:soil\_3.tif

[irrigation]

irrig\_rast:cantons\_irrigues.tif

[dir\_out]

files:Rhone\_interm

results:Rhone\_results

# -------------------------

# 1st step : hru-delin\_init

# -------------------------

[surface]

#selection: total -> full dem

# polygon -> polygon: name of the shapefile

# coords -> give the coords upper left (west and north) and lower right (east and south)

selection:total

polygon:

west:

north:

east:

south:

[demfill]

#

# if demfill = yes : depressionless DEM will be generated

# no : no action on input DEM

#

demfill:yes

#

# if rules\_auto\_\* = yes : rules will be calculated by the module

# if no : fill the corresponding file (reclass\_default\_rules\_\*)

#

[reclass\_dem]

rules\_auto\_dem:yes

step\_dem:500

[reclass\_slope]

rules\_auto\_slope:yes

[reclass\_aspect]

rules\_auto\_aspect:yes

[basin\_min\_size]

# number of pixels

size=500

# ---------------------------

# 2nd step : hru-delin\_basins

# ---------------------------

[auto\_relocation]

to\_do:yes

# first rule

surface\_tolerance\_1=10

distance\_tolerance\_1=20

# second rule

surface\_tolerance\_2=30

distance\_tolerance\_2=5

gauge\_area\_col\_name=S\_BH

# unit = 1 : m , = 2 : km

gauge\_area\_unit=2

#relocated\_gauges:/home/chris/DONNEES/Rhone/gauges\_def\_fil\_500.shp

relocated\_gauges:

[for\_watershed\_id]

# column name of the gauge attribute (attribute type must be numeric)

# used for identification of watersheds

col\_name=CODE

# ---------------------------

# 3rd step : hru-delin\_hrugen

# ---------------------------

[hrus\_min\_surface]

#

surface=125

#

# MNT-derived layers to be integrated in the overlay operation

#

[layer\_overlay]

dem:x

slope:x

aspect:x

# --------------------------------

# 4th step : hru-delin\_parms\_J2000

# --------------------------------

[topology]

dissolve\_cycle:y

# Annexe 5 : lancement de HRU-delin sous Pénélope

## Pénélope, connexion sur Pénélope

Pénélope est une machine linux accessible aux chercheurs d’IRSTEA (linux, windows ou MAC) après la création d’un compte par le service informatique (en 2016 : [claire.rambeau@irstea.fr](mailto:claire.rambeau@irstea.fr) ou [antoine.gallavardin@irstea.fr](mailto:antoine.gallavardin@irstea.fr)). Si vous souhaitez faire tourner HRU-delin sur Pénélope, demandez aussi un accès au répertoire « modèles » de Pénélope.

Une fois votre compte créé, se connecte sur Pénélope via :

* + Pour les linuxiens : dans un terminal : ssh –X nom-de-compte@penelope ; puis mot-de-passe
  + Pour les windowsiens, via un logiciel comme Putty (à installer).

## HRU-delin sous Pénélope

Le répertoire /home/modeles sous Pénélope est dédié aux modèles d’Irstea, dont HRU-delin.

Pour faire tourner HRU-delin sous Pénélope, la procédure est la suivante :

* Sous /home/modeles, créer un répetoire à son nom.

*Ex : mkdir claire.*

* Sous ce répetoire (ex : claire), exécuter le script d’installation de HRU-delin : hru-delin-prep-cas-v4.sh (en référence à la version utilisée de HRU-delin : v4).

*Utilisation : taper dans la console de commande*

*hru-delin-prep-cas-v4.sh ou hru-delin-prep-cas-v4.sh nom\_du\_cas*

*si le cas existe déjà, il pose la question de l'effacer ou non.*

Cette commande crée sous /home/modeles/claire le répertoire nom\_du\_cas, dans lequel sont rangés les répertoires grass-db et work.

* Aller sous /home/modeles/claire/nom\_du\_cas/work, et
  + Renseigner le fichier de configuration et les fichiers de reclassification si nécessaire (cf Annexe 4 : exemple de fichier de configuration et 6.2.1Discrétisation des altitudes, pentes et orientations)
  + Exécuter successivement hru-delin\_step1.sh, hru-delin\_step2.sh, etc…., comme sur une machine standard.

## Données pour HRU-delin sous Pénélope

Les données standard (= issues du projet MDR) pour faire tourner HRU-delin sur l’Ardèche sont stockées dans /home/modeles/DATA/hru-delin-data-Ardeche

Les utilisateurs d’HRU-delin sur d’autres bassins ou avec d’autres données sont invités à **placer ces données dans un répertoire sous /home/modeles/DATA/** **et à renseigner le ReadMeDATA** pour expliquer d’où viennent ces données.