

# 4.6. BASE DE DONNÉES TOPOGRAPHIQUES COLLABORATIVE POUR LES CAVITÉS PÉRUVIENNES

Ecrit par : Xavier ROBERT

## Introduction

Cet article s'adresse en premier lieu à toutes les personnes actives en exploration et en topographie sur les massifs karstiques péruviens. Mais il peut aussi intéresser les spéléos qui sont intéressés par mettre un place sur un autre système karstique une base de données topographiques similaire. Même si la base de données que nous allons décrire est dédiée à une utilisation avec Therion principalement, cet article ne traite pas de l'utilisation de Therion ni de son intérêt ou sa puissance.

Suite à un travail important de classification des données, transcription des données et dessins topographiques fait par Jean Loup Guyot, j'ai migré l'ensemble des données topographiques sous le logiciel libre de topographie Therion (<https://therion.speleo.sk/>), en la hiérarchisant. L'avantage, c'est qu'elle peut être facilement utilisée dans le cadre d'une étude SIG. Depuis 2016, après chaque séance topographique, je m'occupe du report des données au format Therion qu'on m'envoie, et souvent je dessine aussi la topographie en fonction des carnets topographiques. Cela fonctionne, mais i) demande pas mal de travail et d'aller-retour, ii) le report topographique peut être en conséquences faussé par ma méconnaissance de la zone explorée et topographiée, et iii) implique que je

suis actuellement quasiment la seule personne ressource de cette base de données topographiques, ce qui n'est pas acceptable sur du long terme, d'autant plus que, mon expatriation au Pérou se terminant mi-2020, je risque de moins m'impliquer dans le maintien et la gestion de cette base de données topographiques.

Je propose donc de mettre en place un système de base de données collaborative afin d'alléger le travail et surtout d'ouvrir la possibilité de participer à la maintenance et l'évolution de la base de données topographiques. Par soucis de simplicité, j'ai fait le choix de mettre en ligne cette base de données topographiques sur la plateforme Github (<https://github.com/>) à l'adresse : [https://github.com/robertxa/Mapas\\_Cavernas\\_Peru](https://github.com/robertxa/Mapas_Cavernas_Peru) (DOI : <http://doi.org/10.5281/zenodo.3610615>). Evidemment, comme pour chaque topographie que j'ai travaillée, j'ai mis le travail sous licence libre Créative Commons avec attribution, partage à l'identique et non commerciale (Attribution-ShareAlike-NonCommercial) : <http://creativecommons.org/licenses/by-nc-sa/4.0/>.

# Description de la base de données

La base de données topographiques a pour buts :

- 1) de construire les topographies de chaque cavité topographiée au Pérou, et si possible de hiérarchiser les données,
- 2) d'exporter la topographie de n'importe quel sous-réseau appartenant à une cavité plus importante,
- 3) et enfin de construire des plans, cartes et modèles géo-référencés en trois dimensions de chaque système hydrogéologique sélectionné à la demande.

J'ai donc tenté de structurer la base de données de façon d'une part répondre aux principaux buts évoqués ci-dessus, et d'autre part à faciliter au maximum son utilisation et sa mise à jour.

## 1. Fichiers contenus dans la base de données

La base de données topographiques a été pensée pour une utilisation avec le logiciel libre Therion parce qu'il permet entre autre le travail sur des systèmes hydrokarstologiques complets. En conséquence, cette base de données contient tous les fichiers nécessaires à la compilation et la construction des différentes topographies avec Therion :

- Les fichiers de données topographiques ainsi que leurs jointures, si existantes (fichiers avec l'extension .th)
- Les fichiers de dessins des plans et des coupes (fichiers avec l'extension .th2)
- Les fichiers de compilation qui permettent la compilation des données topographiques et des dessins (fichiers avec l'extension .thconfig)

De plus, les topographies ante-2016 avaient été faites au format numérique en utilisant Visual Topo. J'ai gardé en archive toutes les données Visual Topo qui ont été produites avant le passage à Therion (fichiers avec l'extension .tro).

Enfin, à la racine, j'ai rajouté un fichier de définition de symboles et de configurations qui est appelé par chacun des fichiers de compilation .thconfig (fichier config.thc).

Tous ces fichiers là sont des fichiers textes, qui peuvent être édités avec un éditeur de texte digne de ce nom (i.e. Notepad++ sous Windows, TexWrangler/BEdit, Atom, Brackets sous n'importe quel OS). Des extensions pour colorer la syntaxe Therion peuvent être rajoutées.

Enfin, vous trouverez aussi un fichier caché .gitignore (décrit dans le paragraphe 4. Petit Tutoriel pour utiliser GitHub) qui définit les fichiers qui peuvent être produits ou rajoutés dans les dossiers de la base de données, mais qui ne seront pas synchronisés avec la base de données

en ligne. Pour des raisons de taille de fichiers et de la base de données, j'ai volontairement exclu tous les fichiers :

- d'export de Therion (les .pdf, .log, .kmz, .lox, .3d, .html, .sql, .shp...) ; ils sont facilement régénérés par compilation ;
- images, que ce soit les scans des carnets topos ou des dessins à la main ayant servis à dessiner les topographies ;
- qui ont été utilisés pour produire par exemple les modèles numériques de terrain (MNT) ;
- Les anciens fichiers Adobe Illustrator qui sont lourds.

## 2. Structure de la base de données topographiques

La connaissance des réseaux karstiques péruviens est très parcellaire. J'ai donc structuré la base de données d'abord par appartenance géographique (région, district, commune), puis lorsque je le pouvais, en systèmes hydrologiques en regroupant pour chaque système les différentes cavités qui le compose. Et évidemment, tout au bout de l'arborescence, j'ai créé un dossier par cavité.

Pour les dossiers ou cavités complexes, afin de garder des dossiers simples, j'ai rajouté :

- un dossier Outputs qui contient les sorties de la compilation Therion, à savoir les topographies en pdf, les shapefiles, les modèles 3D, les listes,...
- parfois des dossiers data-th et data-th2 qui contiennent respectivement les données topographiques et les dessins.

Si vous modifiez le nom des dossiers, ou si vous en rajoutez, il est fondamental de ne pas inclure d'espaces ou de caractères accentués/spéciaux, sous peine d'avoir des difficultés de compilation parce que les chemins des fichiers seront mal reconnus.

## 2.1. A la racine

En plus des fichiers .gitignore et config.thc évoqués ci-dessus, il y a plusieurs fichiers :

- README.rst : ce fichier correspond à la description du projet, au format rst.
- Todo.txt : Fichier texte qui décrit le travail qu'il reste à faire sur la base de données pour l'améliorer.

Ce dossier racine contient aussi un dossier par région péruvienne. Chaque dossier « région » est divisé par districts, eux même divisés par municipalité. C'est dans ces derniers où nous trouvons un dossier par cavité connue et topographiée. Si un système hydrogéologique est défini, il est contenu dans un dossier spécifique contenant lui même un dossier par cavités.

## 2.2. Dans chaque dossier cavité

Soit xx le nom de la cavité en question. Dans chaque dossier cavité « XX », nous trouvons (figure 1) :

- xx.th : les fichiers .th stockent les données topographiques en elle-même (voir le wiki de Therion pour une description de la structure de ce type de

fichier). Ces fichiers peuvent être dans un dossier data-th/ pour simplifier le dossier de la cavité. Lorsqu'il y a plusieurs fichiers xx.th, le fichier xx-tot.th permet d'appeler tous les fichiers topographiques, de les joindre entre eux, et de joindre les dessins associés.

- xx.th2 : Ce sont les fichiers de dessins. La plupart du temps, xx.th2 correspond au plan, et xx-coupe.th2 à la coupe développée. Pour les cavités complexes, ces fichiers de dessins peuvent être dans un dossier data-th2/.

- xx-maps.th : Ce fichier permet de définir les différentes maps qui permettent dans le xx.thconfig de sélectionner la partie de la topographie à exporter.

- xx.thconfig : le fichier contenant les commandes de compilation Therion adaptées à la cavité xx.

Dans le cas des systèmes hydrogéologiques, il peut y avoir un dossier DEM contenant les MNT utilisés pour construire les modèles 3D des systèmes.

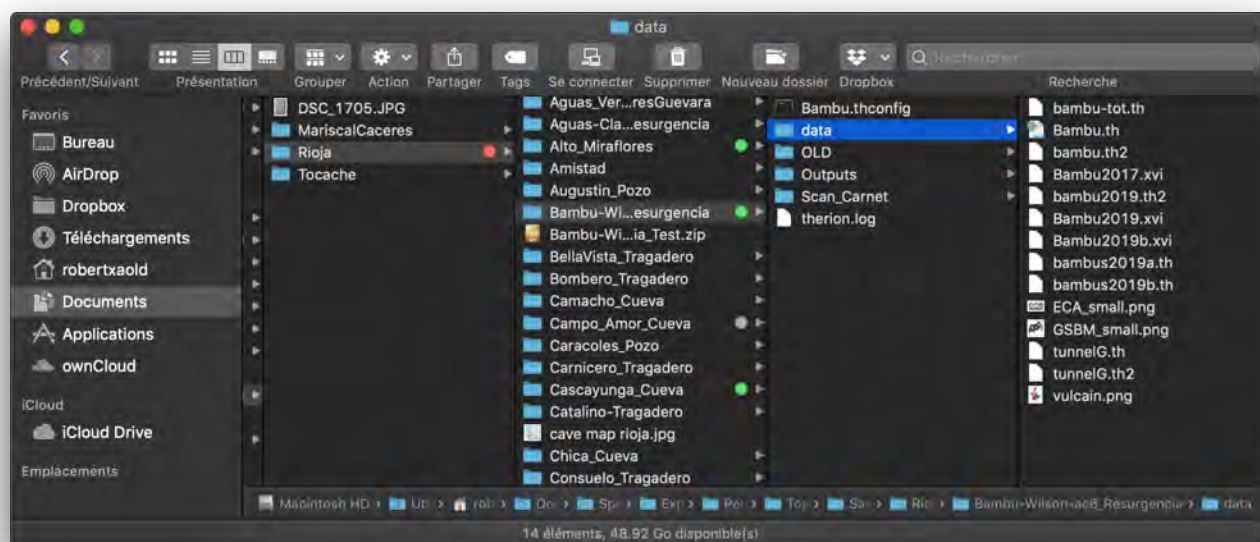


Figure 1 : Exemple de structuration avec le dossier correspondant à la cavité « Cueva de los Bambus » dans la région San Martín.

# Prérequis

Pour pouvoir utiliser cette base de données topographiques et participer à sa maintenance et évolution, certains points sont prérequis, que j'expose ci-dessous.

## 1. Connaissances de base du logiciel Therion

Cet article n'a pas pour but d'expliquer le fonctionnement du logiciel libre Therion, mais comme la base de données est construite autour de ce logiciel, il est nécessaire d'avoir quelques bases pour pouvoir utiliser les données. Vous trouverez toutes les informations nécessaires sur le site internet du logiciel (<https://therion.speleo.sk/>) et sur son wiki (<https://therion.speleo.sk/wiki/doku.php> ; en anglais mais très complet). Afin d'aider à l'utilisation du logiciel, j'ai aussi écrit des templates qui peuvent être utiles et que j'ai mis en ligne sur <https://github.com/robertxa/Th-Config-Xav>.

Normalement, chaque cavité peut être compilée avec le logiciel graphique xtherion ou dans le terminal ou la fenêtre console par la commande `therion xx.thconfig` où `xx` correspond au nom de la cavité (voir le contenu du dossier dans lequel vous travaillez et la documentation Therion).

Pour visualiser les modèles en trois dimensions (fichiers avec l'extension `.lox`), il y a plusieurs possibilités :

- En installant la suite libre Therion, vous installez aussi le logiciel libre Loch qui est le visualisateur 3D de Therion. Dans ce logiciel, une fois un modèle 3D ouvert, il est possible d'importer d'autres modèles de cavités proches avec la fonction Fichier → Importer.
- Sinon, l'autre possibilité est d'ouvrir un des modèles 3D sur la page internet du club Vulcain (basé sur l'application CaveView , <https://aardgoose.github.io/CaveView.js/>) ou directement sur le site internet du développeur du visualisateur et, une fois le modèle 3D chargé, effectuer un glissé-déposé du modèle que vous voulez visualiser (<https://www.groupe-speleo-vulcain.com/explorations/topographies-3d/>).

## 2. Connaissances de base de la plateforme GitHub :

GitHub (<https://github.com/>) est un service web d'hébergement et de gestion de projets qui utilise le logiciel de gestion open-source de version Git. Alors que le système traditionnel open-source amène chaque contributeur à télécharger les sources du projet et à proposer ensuite ses modifications à l'équipe du projet, GitHub fonctionne sur le principe du fork (branches) par défaut : toute personne forkant le projet devient publiquement de facto le chef de son projet portant le même nom que l'original.

### 2.1. Le fork ou la branche

Une branche correspond à une copie du projet principal (« master ») à un moment donné, et qui peut-être modifiée de manière autonome sans impacter le projet initial. Ceci permet de travailler à plusieurs en parallèle, sans pour autant altérer le projet initial. Après avoir terminé de travailler sur une branche, il est possible d'en faire un projet autonome, ou mieux, de proposer de fusionner les modifications de la branche en question avec le projet initial. C'est ce qu'on appelle proposer une « pull request ». Le ou les propriétaire(s) du projet « master » accepteront (ou non !) les modifications proposées, une à une.

C'est ce qui fait la force de la plateforme Github.

### 2.2. La gestion de versions

GitHub est basé sur le logiciel Git qui est un logiciel de contrôle de version. Cela veut dire qu'il gère les modifications d'un projet sans écraser n'importe quelle partie du projet. En pratique, Git permet d'enregistrer spécifiquement des modifications différentes d'un projet par différentes personnes sans écraser le travail des autres personnes. Plus tard, nous pouvons choisir de fusionner tel ou tel partie sans perdre le travail qui a été fait. En plus, s'il le faut, nous pouvons revenir en arrière parce que Git conserve une copie de tous les changements.

### 2.3. Pourquoi ce choix ?

J'ai choisi d'utiliser cette plateforme d'une part pour les conditions d'utilisations (on conserve la propriété complète de tous les projets déposés), l'hébergement gratuit, l'accessibilité à tous et d'autre part pour les facilités d'utilisation de Git via la plateforme. Via un logiciel multiplateforme d'aide à l'utilisation de Git pour les « non-informaticiens ».

En effet, il est possible d'interagir avec les projets sur la plateforme GitHub uniquement avec les commandes en ligne de l'outil Git, mais le logiciel GitHub Desktop (multiplateforme, <https://desktop.github.com/>) facilite grandement le processus. Je vous conseille de l'installer pour travailler sur la base de données.

Il existe d'autres plateformes de ce type (comme Gitlab par exemple), mais un peu plus difficile d'accès pour les non-informaticiens. C'est pourquoi j'ai fait ce choix, mais il serait tout à fait possible de migrer le projet GitHub (actuellement racheté par Microsoft) vers une plateforme libre.

Comme l'utilisation de l'interface GitHub n'est pas forcément intuitive pour tout le monde, je propose ci-dessous un petit tutoriel pour l'utiliser dans le cadre de la maintenance (et du travail collaboratif) sur la base de données des topographies du Pérou.

## Petit tutoriel pour utiliser Github

La plateforme GitHub est en anglais. Les termes utilisés sont simples et compréhensibles, mais dans la suite, je tente tout de même une traduction des termes les plus importants pour faciliter la compréhension aux personnes fâchées avec la langue de Shakespeare. Il existe beaucoup de documentation sur internet, à la fois en anglais et en français, ne pas hésiter à la chercher si besoin. Et évidemment, comme pour tout, il faut lire toutes les lignes, même les petites !

### 1. Comment télécharger la base de données topographique et jouer avec

Télécharger la base de données pour usage personnel est simple : sur la page internet du projet (repository en langage Github) [https://github.com/robertxa/Mapas\\_Cavernas\\_Peru](https://github.com/robertxa/Mapas_Cavernas_Peru), cliquez sur le bouton vert « Clone or download » (= cloner ou télécharger), puis sur « Download Zip » (télécharger le zip). Il ne vous reste qu'à décompresser l'archive zip dans le dossier où vous voulez travailler, puis à jouer avec !

### 2. Comment travailler collaborativement sur la base de données, la mettre à jour et proposer des modifications

Pour travailler collaborativement, télécharger la base de données sur son ordinateur personnel ne suffit pas. Il est nécessaire de se créer un compte GitHub, afin de pouvoir profiter de toutes les options de Git et GitHub. Je décris ci-dessous comment créer un tel compte, puis comment faire un fork de la base de données, comment mettre à jour ce fork et enfin comment proposer une pull request à un des administrateurs de la base de données pour qu'il révise et intègre les modifications.

Je vous conseille aussi de télécharger le logiciel Github Desktop (<https://desktop.github.com/>).

#### 2.1 Créer un compte personnel dans GitHub

Sur la page principale de GitHub (<https://github.com/>), cliquez en haut à droite sur « Sign in » (= s'enregistrer), puis, sur la nouvelle page « create a new account » (= créer un nouveau compte). Remplir les champs demandés et suivre les instructions.

Le compte ainsi créé se trouvera à la page <https://github.com/Username/> où Username est à remplacer par votre identifiant personnel (par exemple, mon Username est robertxa, et mon compte se trouve sur <https://github.com/robertxa/>).

Une fois le compte créé, pour se connecter, accéder à n'importe quelle page de Github (la page principale, ou la page d'un des projets existant), et cliquez en haut à droite sur « Sign in » (= s'enregistrer). Puis, sur la nouvelle page, entrez l'username (=identifiant défini lors de l'ouverture du compte) ou l'adresse email utilisée pour ouvrir le compte, et le Password (=mot de passe défini lors de l'ouverture du compte).

#### 2.2. Créer une « branche » et la mettre à jour

Avant tout, lancez le logiciel Github Desktop, et connectez vous via ce logiciel à votre compte Github. Ensuite, cliquez sur File → Clone Repository et entrer l'adresse web de la base de données. Avant d'appuyer sur le bouton « cloner », vous devez renseigner votre dossier local (sur votre ordinateur) dans lequel sera clonée la base de données.

Une fois le clonage effectué, vous aller pouvoir travailler directement dans la base de données, dans le dossier local que vous avez défini au moment du clonage. Chaque fois que vous aller effectuer une modification, elle apparaîtra dans le logiciel Github Desktop. Pour mettre à jour votre repository Github (votre branche), il faut cliquez sur « Commit » (il faut renseigner obligatoirement le nom du commit), puis sur Publish. Les modifications seront copiées sur le serveur.



### 2.3. Proposer une « pull request »

Une fois votre contribution prête, il faut l'ajouter au projet initial. Pour cela, il faut demander à ajouter vos modifications et/ou ajouts à la base de données principale. Ceci s'effectue par ce qui s'appelle une « pull request ».

Pour cela, vous pouvez l'effectuer, soit via l'interface en ligne de Github, soit via le logiciel Github Desktop. Sur votre compte, allez dans votre projet et cliquez sur Create Pull request. Dans le logiciel, sélectionner le projet (votre branche), puis aller dans Branch → Create Pull Request. Les deux méthodes mènent à la page permettant de renseigner les différents champs et de finaliser la démarche.

### 2.4. Mettre à jour son propre repository via le « Master »

Une fois que la base de données principale a été mise à jour, vous pouvez mettre à jour votre branche en cliquant dans le logiciel Github Desktop sur Repository → Pull.

## 3. Petit mémo des commandes en ligne Git pour les Geeks irréductibles

Sur internet, il y a de nombreux tutoriels pour utiliser Git, je ne vais pas les reprendre, mais voilà au cas où quelques commandes Git du terminal importantes :

- `git init <dossier>` : transformer n'importe quel dossier en un dépôt Git.
- `git clone https://github.com/robertxa/Topographies-Samoens\_Folly` : cloner la base de données sur son ordinateur local.

- `git status` : indique les fichiers que vous avez modifiés.
- `git diff` : indique concrètement ce que vous avez changé.
- `git add nomfichier1 nomfichier2` : ajouter les fichiers à la liste de ceux devant faire l'objet d'un commit.
- `git commit -a` : pour committer tous les fichiers listés dans `git status` dans les colonnes « changes to be committed » et « Changed but not upadted ».
- `git log` : affiche la liste des commits.
- `git reset HEAD/HEAD^/HEAD^^/HEAD~2/N°_de_commit` : indique à quel commit on souhaite revenir (HEAD = dernier commit ; HEAD^ = avant-dernier commit ; HEAD^^ = HEAD~2 = avant avant dernier commit ; N°\_de\_commit).
- `git reset --hard HEAD/HEAD^/HEAD^^/HEAD~2/N°_de_commit` : annule les commits et perd tous les changements.
- `git pull` : téléchargement des nouveautés depuis le serveur.
- `git push` : envoyer vos commits sur le serveur.
- `git branch` : pour voir toutes vos branches.
- `git branch <nom_branche>` : créer une branche.
- `git branch-d <nom_branche>` : supprimer une branche.
- `git checkout <nom_branche>` : aller dans la branche nom\_branche.
- `git merge` : fusionner les changements.



## Perspectives

Et voilà, vous êtes prêts à utiliser la base de données des topographies des cavernes péruviennes, vous avez tout en main, à vous de jouer. Pour l'instant, cette base de données est sur mon compte GitHub. C'est à discuter, mais il serait possible d'ouvrir un compte ECA par exemple, avec deux ou trois personnes administratrices. Aussi, à terme, il serait peut-être intéressant de lier cette base de données topographiques avec la base de données des fiches cavités mise en place et maintenue par Jean Loup Guyot (<https://cuevasdelperu.org/>).

Pour ceux qui sont intéressés, j'ai créé un projet similaire pour les cavités du massif du Folly à Samoëns (France) ([https://github.com/robertxa/Topographies-Samoens\\_Folly](https://github.com/robertxa/Topographies-Samoens_Folly)). N'hésitez pas à participer à ces projets, à m'envoyer des commentaires ou des propositions de modifications si besoin, et surtout, n'hésitez pas à me demander de plus amples détails et/ou aide si besoin !