



# LA BIBLE *Therion*

Stacho Mudr'ak  
Martin Budaj

suivie de

- **La topographie avec *Therion***  
(par Xavier Robert, Groupe Spéléologique Vulcain)
- ***Therion* pour les spéléos purs et durs**  
(Wiki *Therion*)
- **Le manuel de *Therion***  
(Wiki *Therion*)
- **Un tutoriel *Therion***  
(par Footleg)

V. 4.2 traduction et mise en page : Dominique Ros, 2021

Therion est un logiciel protégé par le droit d'auteur. Distribué sous licence publique générale GNU.

Copyright © 1999–2021 Stacho Mudrák, Martin Budaj

Ce livre décrit *Therion* version 5.5.7+5fc5b63 (2021-06-06).

Contributions au code de Matěj Pích, Olly Betts, Marco Corvi, Vladimir Georgiev, Georg Pacher et Dimitrios Zachariadis.

Nous remercions

Martin Sluka, Ladislav Blazek, Martin Heller, Wookey, Olly Betts et tous les utilisateurs pour leurs commentaires, leur soutien et leurs suggestions.

Traductions (%):

Langue	XTherion	Map header	Loch	Traduit par
bg	81	77	100	Alexander Yanev, Ivo Tachev, Vladimir Georgiev
ca	–	80	–	Evaristo Quiroga
cs	83	75	–	Ladislav Blažek Roger Schuster, Georg Pacher, Benedikt
de	97	74	–	Hallinger
el	80	69	–	Stelios Zacharias
en[ GB  US]	72	100	100	Stacho Mudrák, Olly Betts, Rodrigo Severo
es _ _	100	79	–	Roman Muñoz, Evaristo Quiroga
fr	99	79	–	Eric Madelaine, Gilbert Fernandes
it	100	99	100	Marco Corvi
mi	–	71	–	Kyle Davis, Bruce Mutton
pl	–	71	–	Krzysztof Dudziński
pt[ BR  PT]	–	78	–	Toni Cavalheiro, Rodrigo Severo
ru _ _	97	77	–	Vasily V. Suhachev, Andrey Kozhenkov
sk	96	78	96	Stacho Mudrák
sl	100	80	96	Fatos Katallozi
sq	80	69	–	Fatos Katallozi
zh	81	71	–	Zhang Yuan Hai, Duncan Collis

L'illustration de couverture montre un croquis de la salle de *Hrozný kameňolom* dans la « Grotte des chauves-souris mortes » (Cave of Dead Bats) en Slovaquie et la topographie de celle-ci produite par *Therion*.

# Tables des matières

Tables des matières.....	3
Introduction .....	6
Pourquoi <i>Therion</i> ?.....	6
Caractéristiques.....	7
Logiciels requis .....	7
Installation.....	8
Environnement de travail.....	8
Comment cela fonctionne-t-il ? .....	9
Première utilisation.....	10
<b>Création de fichiers de données .....</b>	<b>10</b>
Les bases .....	10
Types de données.....	11
Systèmes de coordonnées .....	12
Déclinaison magnétique automatique ou manuelle.....	13
Format des données.....	13
Attributs personnalisés .....	34
<i>XTherion</i> .....	34
<i>XTherion</i> — module éditeur de texte.....	35
<i>XTherion</i> — module éditeur de dessin.....	35
Outils additionnels .....	37
Raccourcis clavier et souris dans l'éditeur de dessins .....	38
Penser <i>Therion</i> .....	40
Comment entrer une ligne de cheminement ?.....	40
Comment dessiner des topographies ?.....	41
Comment créer des modèles ? .....	42
<i>Therion</i> en profondeur.....	42
Comment le dessin est-il construit ?.....	42
<b>Traitement des données.....</b>	<b>44</b>
Fichier de configuration .....	44
Faire fonctionner <i>Therion</i> .....	58
<i>XTherion</i> —compiler.....	60
<b>Qu'obtenons-nous ? .....</b>	<b>60</b>
Les fichiers d'information.....	60
Fichier Log .....	60
<i>XTherion</i> .....	60
Export SQL.....	60

Listes de cavités, surveys (topographies), continuations (points d'interrogation).....	62
Plans 2D.....	62
Plans à imprimer .....	62
Maps for GIS.....	63
Topographies pour applications spécifiques.....	63
Modèles 3D .....	63
Loch .....	63
Mise en page des dessins en PDF.....	63
Mise en page en mode Atlas.....	63
Mise en page en mode topographie (cartographie).....	68
Personnalisation des étiquettes de texte .....	70
Nouveaux symboles topographiques.....	70
Symboles de points .....	71
Symboles de lignes.....	72
Symboles de surfaces.....	72
Symboles spéciaux .....	73
<b>Annexe .....</b>	<b>73</b>
Compilation.....	73
Démarrage rapide .....	<b>Erreur ! Signet non défini.</b>
Le guide du Hacker.....	75
Variables d'environnement.....	77
Fichiers d'initialisation.....	77
<i>Therion</i> .....	<b>Erreur ! Signet non défini.</b>
<i>XTherion</i> .....	81
Limitations.....	81
Exemple de données.....	82
Historique.....	83
Le futur .....	84
Général.....	84
Dessins 2D .....	84
Modèles 3D .....	84
<i>XTherion</i> .....	84
Loch .....	84
Labyrinth .....	85
Études de cas.....	86
Dessiner une topo avec <i>Therion</i> .....	86
Listing de cavités .....	86

Symboles de surfaces .....	87
Types de surfaces <i>Therion</i> .....	87
Morphing de croquis .....	88
Image après morphing : .....	88
Parois après morphing .....	89
Image après morphing : .....	89
Image après morphing et insertion de .....	90
Contrôle de coupe développée .....	90
Les stations sur des segments en coupe développée .....	93
Lecture différée des cartes superposées en offset .....	94
Importer des fichiers <i>Survex</i> .3d .....	96
Utilisation des topographies spécifiées dans les fichiers .th.....	96
Création de topographies inexistantes .....	<b>Erreur ! Signet non défini.</b>
Ignorer les préfixes de stations .....	98
Gérer des projets complexes.....	98
Conclusion .....	100
Indicateurs de points d'interrogation (continuation) sur la ligne de cheminement.....	100
Indicateurs de points d'interrogation (continuation) sur le dessin .....	102
Utiliser des types de symboles définis par l'utilisateur.....	103

## NE LAISSEZ PERSONNE IGNORANT LA GEOMETRIE ENTRER ICI

ΑΓΕΩΜΕΤΡΗΤΟΣ ΜΗΔΕΙΣ ΕΙΚΙΤΩ

\* Inscription sur l'entrée de l'Académie de Platon, IVe siècle av. \*

### **Introduction**

*Therion* est un outil de topographie souterraine. Son but est d'aider à

- archiver les données topographiées sur ordinateur sous une forme aussi proche que possible des notes et des esquisses d'origine et les récupérer de manière flexible et efficace;
- dessiner et actualiser de beaux plans et coupes ;
- créer un modèle 3D réaliste de la grotte.

Il fonctionne avec les systèmes d'exploitation Unix, Linux, MacOS X et Win32. Le code source et le programme d'installation de Windows sont disponibles sur la page Web de *Therion* (<https://Therion.speleo.sk>).

*Therion* est distribué sous licence **GNU General Public License**.

### **Pourquoi *Therion* ?**

Dans les années 90, nous avons beaucoup pratiqué la spéléologie et la topographie. Il existait certains programmes informatiques affichant des vues de cheminement et des stations après la fermeture de la boucle et l'élimination des erreurs. Ceux-ci ont été d'une grande aide, en particulier pour les systèmes de cavités de grande taille et complexes. Nous avons utilisé l'un d'eux, TJKPR, comme couche de fond avec le calque des stations pour dessiner ensuite des topographies à la main. Après avoir achevé un vaste atlas de la « Grotte des chauves-souris » de 166 pages au début de 1997, nous avons rapidement eu un problème : nous avons trouvé de nouveaux passages reliant des galeries connues et les avons explorés. Après traitement dans TJKPR, les nouvelles boucles ont influencé la position des anciens relevés. La plupart des stations du relevé avaient maintenant une position légèrement différente d'avant en raison de la modification de la répartition des erreurs. Ainsi, nous devons soit dessiner à nouveau tout l'Atlas, soit accepter que l'emplacement de certains endroits ne soit pas précis (dans le cas de boucles d'une longueur d'environ 1 km, il y avait parfois des erreurs d'environ 10 m) et d'essayer de fausser les nouvelles galeries pour les adapter aux anciennes.

Ces problèmes ont persisté lorsque nous avons essayé de dessiner des topographies à l'aide de certains programmes de CAO en 1998 et 1999. Il était toujours difficile d'ajouter de nouveaux relevés sans adapter les anciens aux positions nouvellement calculées des stations topo dans toute la grotte. Nous n'avons trouvé aucun programme capable de dessiner une topographie complexe correctement mise à jour (c'est-à-dire pas seulement des vues avec une enveloppe LRUD), dans laquelle les anciennes parties soient modifiées en fonction des coordonnées connues des stations du relevé le plus récent.

En 1999, nous avons commencé à réfléchir à la création de notre propre programme pour le dessin de topographies. Nous connaissions des programmes parfaitement adaptés à des sous-tâches particulières. Il y avait METAPOST, un langage de programmation de haut niveau pour la description de graphiques vectoriels, *Survex* pour un excellent traitement des plans, et T<sub>E</sub>X pour la compilation des résultats. Nous n'avions qu'à les assembler. À Noël 1999, une version minimaliste de *Therion* fonctionnait pour la première fois. Cela ne comprenait que 32 kB de scripts Perl et de macros METAPOST, mais visait à démontrer que nos idées pouvaient être mises en œuvre.

En 2000-2001, nous avons recherché le format optimal des données d'entrée, le langage de programmation, le concept d'éditeur de dessins interactif et des algorithmes internes avec l'aide de Martin

Sluka (Prague) et de Martin Heller (Suisse). En 2002, nous avons pu sortir la première version de *Therion* réellement utilisable, qui répondait à nos exigences.

## Caractéristiques

*Therion* est une application en ligne de commande. Il traite les fichiers d'entrée, y compris Les topographies 2D, au format texte, et crée des fichiers avec des topos 2D ou un modèle 3D en sortie.

La syntaxe des fichiers d'entrée est décrite en détail dans les chapitres suivants. Vous pouvez créer ces fichiers dans un éditeur de texte brut quelconque tel que *ed* ou *vi*. Ils contiennent des instructions pour *Therion* comme par exemple :

```
point 1303 1004 pillar
```

où *point* est un mot-clé pour le symbole de point, suivi de ses coordonnées et d'une spécification de type de symbole à afficher.

L'édition manuelle de tels fichiers n'est pas facile - en particulier lorsque vous tracez des topos, vous devez penser en termes spatiaux (coordonnées cartésiennes). Il existe donc une interface graphique spéciale pour *Therion* appelée *XTherion*. *XTherion* fonctionne comme un éditeur de texte avancé, un éditeur de topos (où les images sont dessinées de manière totalement interactive) et un compilateur (qui exécute *Therion* à partir de ces données).

Cela peut paraître compliqué, mais cette approche présente de nombreux avantages :

- Il y a une séparation stricte des données et de la visualisation. Les fichiers de données spécifient uniquement où sont les objets, pas ce à quoi ils ressemblent. METAPOST ajoute la représentation visuelle dans les phases ultérieures du traitement des données. (Ce concept est très similaire à la représentation de données XML.)

Cela permet de modifier les symboles de dessin utilisés sans modifier les données d'entrée ou de fusionner plusieurs dessins créés par différentes personnes ayant des styles différents dans un seul dessin avec un jeu de symboles unifié.

Les dessins en 2D sont adaptés à des échelles de sorties spécifiques (niveau d'abstraction, mise à l'échelle non linéaire des symboles et des textes)

- Toutes les données sont relatives aux positions des stations de la topographie. Si les coordonnées des stations de la topographie sont modifiées au cours du processus de fermeture de la boucle, toutes les données pertinentes sont déplacées en conséquence, de sorte que la topographie est toujours à jour.
- *Therion* n'est pas dépendant d'un système d'exploitation, d'un encodage de caractères ou d'un éditeur de fichiers d'entrée particulier; les fichiers d'entrée resteront lisibles par un être humain.
- Il est possible d'ajouter de nouveaux formats de sortie.
- Le modèle 3D est généré à partir des images 2D pour obtenir un modèle 3D réaliste sans entrer trop de données.
- Bien que le support pour WYSIWYG soit limité, vous obtenez ce que vous voulez.

## Logiciels requis

«Un programme doit faire une seule chose, mais le faire bien.» (Ken Thompson) Nous utilisons donc des programmes externes utiles, liés aux problèmes de composition et de visualisation des données. *Therion*

peut alors s'acquitter de sa tâche beaucoup mieux que s'il s'agissait d'une application autonome permettant de calibrer votre imprimante ou votre scanner et d'envoyer un courrier électronique avec vos données d'un seul clic.

*Therion* a besoin de :

- une librairie PROJ
- une distribution TEX. Nécessaire uniquement si vous souhaitez créer des dessins 2D au format PDF ou SVG.
- Tcl / Tk avec *BWidget* et éventuellement l'extension *tklmg*. Il n'est nécessaire que pour *XTherion*.
- des outils typographiques LCDF si vous souhaitez utiliser une configuration facile pour les polices personnalisées dans les dessins PDF.
- convertissez et identifiez les utilitaires à partir de la distribution *ImageMagick*, si vous souhaitez utiliser la déformation des esquisses de topographie.
- *Ghostscript* si vous souhaitez créer des images calibrées à partir de fichiers PDF géoréférencés.

**Le programme d'installation Windows inclut tous les packages requis**, à l'exception de Ghostscript. Lisez l'Annexe si vous voulez compiler vous-même *Therion*.

Pour afficher des dessins et des modèles, vous pouvez utiliser l'un des programmes suivants :

- tout visualiseur PDF ou SVG affichant des images 2D;
- tout SIG prenant en charge les formats DXF ou « Shapefile » pour l'analyse des dessins;
- une visionneuse 3D appropriée pour les modèles exportés dans un format autre que le format par défaut;
- tout client de base de données SQL pour traiter la base de données exportée.

## Installation

Installation à partir des sources (*Therion-5.\*.tar.gz* package):

Le code source est une distribution primaire de *Therion*. Il doit être compilé et installé conformément aux instructions de l'annexe.

Installation pour Windows :

Exécutez le programme d'installation et suivez les instructions. Il installe toutes les dépendances requises et crée des raccourcis vers *XTherion* et *Therion Book*.

## Environnement de travail

*Therion* lit les paramètres du fichier d'initialisation. Les paramètres par défaut devraient fonctionner correctement pour les utilisateurs utilisant uniquement les caractères <sup>1</sup>latins, T<sub>E</sub>X standard et METAPOST.

Si vous souhaitez utiliser vos propres polices pour les caractères latins ou non latins dans les fichiers PDF, modifiez le fichier d'initialisation. Des instructions sur la manière de procéder sont données en annexe.

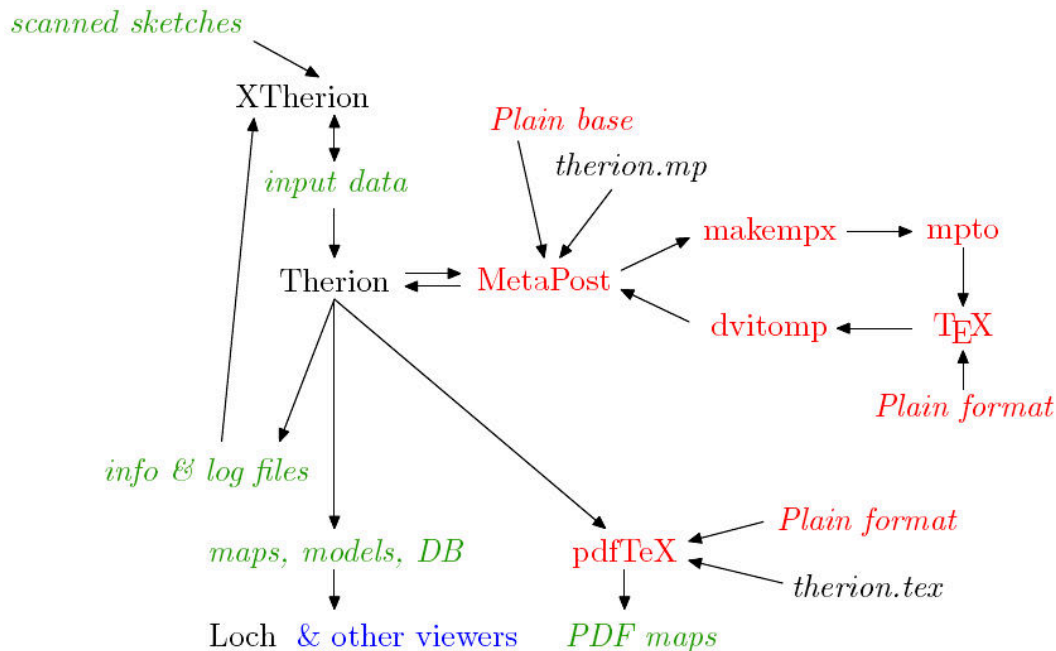
---

<sup>1</sup> Sur le fichier PDF, *Therion* restitue la plupart des caractères accentués sous la forme d'une combinaison d'accent et d'un caractère de base. Certains accents étranges peuvent être omis. Des lettres accentuées précomposées sont incluses pour les langues slovaque et tchèque.



## Comment cela fonctionne-t-il ?

Alors, maintenant que les besoins de *Therion* sont clairs, voyons comment il interagit avec tous ces programmes :



PAS DE PANIQUE ! Lorsque votre système est correctement configuré, la plus grande partie est cachée à l'utilisateur et tous les programmes nécessaires sont automatiquement exécutés par *Therion* de façon transparente.

Pour travailler avec *Therion*, il suffit de savoir que vous devez créer des données d'entrée (la meilleure méthode étant avec *XTherion*), puis exécuter *Therion* et afficher les fichiers de sortie (modèle 3D, dessins, fichier journal) dans le programme approprié.

Pour ceux qui veulent en savoir plus à ce sujet, voici une brève explication de l'organigramme ci-dessus. Les noms de programme sont en police romaine et les fichiers de données en italique. Les flèches indiquent le flux de données entre les programmes. Les fichiers de données temporaires ne sont pas affichés. Signification des couleurs :

- noir ⇒ *Therion* programmes et macros (*XTherion* est écrit en Tcl/Tk, il a donc besoin d'un interpréteur pour fonctionner)
- rouge ⇒ TEX paquets de données
- vert ⇒ fichiers entrants créés par l'utilisateur et fichiers sortants créés par *Therion*

*Therion* effectue lui-même la tâche principale. Il lit les fichiers d'entrée, les interprète, trouve les boucles fermées et distribue les erreurs. Ensuite, toutes les autres données (par exemple, dessins 2D) sont transformées en fonction de la nouvelle position de la station. *Therion* exporte les données des dessins 2D au format METAPOST. METAPOST donne la forme réelle aux symboles de carte abstraits en fonction des définitions de symbole de carte; cela crée beaucoup de fichiers PostScript avec de petits segments de la grotte. Ceux-ci sont lus et convertis dans un format de type PDF, qui forme les données d'entrée pour pdfTeX. pdfTeX effectue ensuite toute la composition et crée le fichier PDF final de la topographie de la grotte.

*Therion* exporte également le modèle 3D (complet ou ligne de cheminement) dans différents formats.

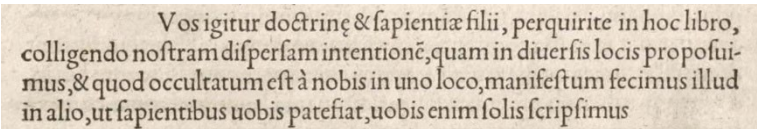
Le cheminement peut être exporté pour un traitement ultérieur dans n'importe quelle base de données SQL.

## Première utilisation

Après avoir expliqué les principes de base de *Therion*, il est judicieux de l'essayer sur des exemples de données.

- Téléchargez les exemples de données à partir de la page Web *Therion* et décompressez-les quelque part sur le disque dur de votre ordinateur.
- Exécutez *XTherion* (sous Unix et MacOS X en tapant «*XTherion*» dans la ligne de commande, vous trouverez sous Windows un raccourci dans le menu Démarrer).
- Ouvrez le fichier "*thconfig*" à partir du répertoire de données de la fenêtre "Compiler" de *XTherion*.
- Appuyez sur «F9» ou «Compiler» dans le menu pour exécuter *Therion* sur les données. Vous obtiendrez des messages de *Therion*, METAPOST et TEX.
- Les fichiers PDF et le modèle 3D sont créés dans le répertoire de données.

De plus, vous pouvez ouvrir des fichiers de données topo (\* .th) dans la fenêtre "Editeur de texte" et des fichiers de données cartographiques (\* .th2) dans la fenêtre "Editeur de carte" de *XTherion*. Bien que le format des données puisse sembler déroutant la première fois, il vous sera expliqué dans les chapitres suivants.



Vos igitur doctrinę & sapientię filii, perquirite in hoc libro, colligendo nostram disperfam intentionē, quam in diuerfis locis proposuimus, & quod occultatum est à nobis in uno loco, manifestum fecimus illud in alio, ut sapientibus uobis patefiat, uobis enim solis scripsimus

*Nous n'avons écrit ce travail que pour vous, enfants de doctrine et d'apprentissage. Examinez ce livre, réfléchissez au sens que nous avons dispersé et rassemblé à nouveau; ce que nous avons caché à un endroit, nous l'avons révélé à un autre, afin que cela puisse être compris par votre sagesse.*

\* Henricus C. Agrippa ab Nettesheym, 1533 \*

## Création de fichiers de données

### Les bases

Les fichiers d'entrée pour *Therion* sont au format texte. Il y a quelques règles sur la forme d'un tel fichier :

- Il existe deux types de commandes. Commandes sur une ligne et commandes sur plusieurs lignes.
- Une commande sur une seule ligne est terminée par un caractère de fin de ligne. La syntaxe de celle-ci est : `commande arg1 ... argN [-option1 valeur1 -option2 valeur2 ...]`

où « arg1 ... argN » sont des arguments obligatoires et les paires commençant par un tiret « -option valeur » sont des options facultatives. Les arguments et les options disponibles dépendent de la commande. Un exemple pourrait être :

```
point 643.5 505.0 gradient -orientation 144.7
```

avec trois arguments obligatoires (au début) et une paire optionnelle « option / valeur » (après le tiret). Les options peuvent n'avoir aucune ou plusieurs valeurs.

Les commandes multilignes commencent de la même manière que les commandes à une seule ligne, mais continuent sur les lignes suivantes jusqu'à la fin de la commande explicite (`endcommande`). Ces lignes peuvent contenir des données ou des options, qui sont appliquées aux données suivantes. Si une

ligne de données commence par un mot réservé à une option, vous devez insérer «!» Devant celle-ci. La syntaxe est :

```
commande arg1 ... argN [-option1 valeur1 -option2 valeur2 ...]
...
optionX valeurX data
...
endcommande
```

Encore une fois, pour une meilleure compréhension, voici un exemple réel avec la commande « `line` » (ligne) :

```
line wall -id walltobereferenced
1174.0 744.5
1194.0 756.5 1192.5 757.5 1176.0 791.0
smooth off
1205.5 788.0 1195.5 832.5 1173.5 879.0
endline
```

Cette commande `line` a un argument obligatoire, un type de ligne (paroi de cavité dans ce cas), suivi d'une option. Les deux lignes suivantes contiennent des données (coordonnées des courbes de Bézier à tracer). La ligne suivante (« `smooth off` » pour lisser) spécifie une option qui s'applique aux données suivantes (c'est-à-dire pas pour la ligne entière, contrairement à l'option `-id` dans la première ligne) et la dernière ligne contient quelques données supplémentaires.

- si la valeur d'une option ou d'un argument contient des espaces, vous devez inclure cette valeur entre " " ou [ ]. Si vous souhaitez insérer un double guillemet " " texte inclus " ", vous devez l'insérer deux fois. Les guillemets sont utilisés pour les chaînes, les crochets pour les valeurs numériques et les mots-clés (*keywords*).
- chaque ligne se terminant par une barre oblique inversée ou antislash (\) est considérée comme continuant sur la ligne suivante, comme s'il n'y avait ni saut de ligne ni barre oblique inversée.
- tout ce qui suit # (dièse), jusqu'à la fin de la ligne, même à l'intérieur d'une commande, est considéré comme un commentaire et est donc ignoré.
- les commentaires multilignes dans un bloc `comment ... endcomment` sont autorisés dans les fichiers de données et de configuration

## Types de données

*Therion* utilise les types de données suivants :

- *keyword* (mot-clé) ▷ une séquence de caractères A-Z, a-z, 0-9 et `_ - /` (ne commençant pas par «-»).
- *ext-keyword* (mot-clé étendu) ▷ mot-clé pouvant également contenir des caractères + \*, ', Mais pas sur la première position.
- *date* (date) ▷ une date (ou un d'intervalle de temps) dans un format spécifique : `YYYY.MM.DD@HH:MM:SS.SS - YYYY.MM.DD@HH:MM:SS.SS` ou '-' pour laisser une date indéterminée.
- *person* (personne) ▷ nom et prénom d'une personne séparés par des espaces. Utilisez '/' pour séparer le prénom et le nom s'il y a plusieurs noms.

- *string* (chaîne) ▷ une séquence de caractères quelconques. Les chaînes peuvent contenir une balise spéciale `<lang: XX>` pour séparer les traductions. Dans les chaînes multilingues, seul le texte compris entre `<lang: XX>` (où XX est la langue sélectionnée dans le fichier d'initialisation ou de configuration) et la balise `<lang: YY>` suivante est affiché à la sortie. Si aucune correspondance n'est trouvée, tout ce qui précède toute balise `<lang: ZZ>` est affiché.
- *units* ▷ unités de longueur prises en charge : mètre [s], centimètre [s], pouce [es], pieds [s], mètre [s] (aussi m, cm, in, ft, yd). Unités d'angle prises en charge : degré [s], minute [s] (également degré, min), degré [s], mil [s], pourcent[age] (clinomètre uniquement). Une valeur en degrés peut être entrée en notation décimale (*x.y*) ou en notation spéciale pour les degrés, minutes et secondes (*deg [: min [: sec]]*).

## Systèmes de coordonnées

*Therion* prend en charge les transformations de coordonnées dans les systèmes de coordonnées géodésiques. Vous pouvez spécifier l'option `cs` (CS pour **C**oordinate **S**ystem) dans les objets `centreligne`, `surface`, `import` et `layout` (ligne de cheminement, surface, importation et mise en page), puis saisir des données `XY` dans un système donné. Vous pouvez également spécifier la sortie `cs` dans le fichier de configuration.

Si vous ne spécifiez pas de `cs` dans votre jeu de données, il est supposé que vous travaillez dans un système de coordonnées local et aucune conversion n'est effectuée. Si vous spécifiez `cs` n'importe où dans les données, vous devez le spécifier pour toutes les données de géolocalisation (`fix`, `origin in layout` etc.).

`cs` s'applique à toutes les données d'emplacement suivantes jusqu'à ce qu'un autre `cs` soit spécifié ou sinon jusqu'à la fin de l'objet en cours, selon celui qui est indiqué en premier.

Les systèmes de coordonnées suivants sont pris en charge :

- `UTM1 – UTM60` ▷ Universal Transverse Mercator pour l'hémisphère nord et dans une zone donnée, WGS84 datum.
- `UTM1N – UTM60N` ▷ idem à `UTM1 – UTM60`
- `UTM1S – UTM60S` ▷ UTM pour l'hémisphère sud, WGS84 datum.
- `lat-long`, `long-lat` ▷ latitude (N positive, S négative) et longitude (E positive, W négative) dans un ordre donné en degrés (deg[:min[:sec]] autorisé), WGS84 datum. Par défaut, non pris en charge en sortie.
- `EPSG:<nombre>` ▷ La plupart des systèmes de coordonnées EPSG. Presque tous les systèmes de coordonnées utilisés dans le monde ont leur propre numéro EPSG. Pour trouver le numéro de votre système, voir le fichier `extern/proj4/nad/epsg` dans la distribution de la source *Therion*.
- `ESRI:<nombre>` ▷ Similaire à EPSG, mais standard ESRI.
- `JTSK`, `iJTSK` ▷ Système tchécoslovaque S-JTSK utilisé depuis les années 1920 avec les axes sud et ouest (JTSK) et sa version modifiée avec les axes orientés est et nord et les nombres négatifs (iJTSK). JTSK n'est pas pris en charge en sortie (mais iJTSK lui l'est).
- `JTSK03`, `iJTSK03` ▷ nouvelle réalisation S-JTSK introduite en Slovaquie en 2011.

- `OSGB` : <H, N, O, S or T><A-Z except I> ▷ Grille nationale du British Ordnance Survey.
- `S-MERC` ▷ la projection sphérique de Mercator, utilisée par divers sites de cartographie en ligne.

### Déclinaison magnétique automatique ou manuelle

*Therion* intègre le modèle de champ géomagnétique Terre IGRF<sup>2</sup>, valable pour la période 1900-2020. Il est utilisé automatiquement pour le calcul de la déclinaison magnétique si la cavité est située dans l'espace avec une station `fix` (géolocalisée) utilisant l'un des systèmes de coordonnées géodésiques pris en charge en même temps que la commande `date` des lignes de cheminement. La déclinaison calculée est alors répertoriée dans le fichier journal pour plus d'informations (LOG file).

Si l'utilisateur définit une `declination` (déclinaison magnétique) spécifique pour le cheminement, ce choix a priorité sur le calcul automatique.

### Format des données

La syntaxe des fichiers d'entrée est expliquée dans la description des commandes individuelles. L'étude des exemples de fichiers distribués avec *Therion* vous aidera à comprendre. Voir aussi un exemple en annexe.

Chacune des sections suivantes décrit une commande *Therion* de la façon suivante :

*Description* : notes concernant cette commande.

*Syntaxe* : description détaillée de la syntaxe utilisant le vocabulaire adéquat écrit en bleu.

*Contexte* : spécifie le contexte dans lequel cette commande est autorisée. Par exemple, le contexte de relevé topo (`survey`) signifie que la commande doit être entourée par la paire `survey ... endsurvey`. Le contexte de segment de calque `scrap` signifie que la commande doit être incluse dans la paire `scrap ... endscrap`. Le *Contexte* : `all` signifie que la commande peut être utilisée n'importe où.

*Arguments* : liste des arguments obligatoires avec leurs explications.

*Options* : liste des options utilisables.

*Command-like Options* : options pour les commandes multilignes, qui peuvent être spécifiées parmi les lignes de données.

#### 'encoding'

*Description* : définit l'encodage du fichier d'entrée. Cela permet d'utiliser d'autres caractères que les caractères ASCII dans les fichiers d'entrée.

*Syntaxe* : `encoding <nom-codage>`

*Contexte* : Ce devrait être la toute première commande du fichier.

*Arguments* :

---

<sup>2</sup> Voir <https://www.ngdc.noaa.gov/IAGA/vmod/>

- `<nom-codage>` ▸ pour voir une liste de tous les noms de codage pris en charge, exécutez *Therion* avec l'option `--print-encodings`. Les codages «UTF-8» (Unicode) et «ASCII» (7 bits) sont toujours pris en charge.

### 'input'

*Description* : insère le contenu d'un fichier à la place de la commande. L'extension par défaut est '.th' et peut être omise. Pour une compatibilité optimale, utilisez les chemins relatifs et les barres Unix "/" (*slash*), et non les barres obliques inverses Windows "\" (*antislash*), comme séparateurs de répertoires.

*Syntaxe* : `input <nom-du-fichier>`

*Contexte* : `all`

*Arguments* :

- `<nom-du-fichier>`

### 'survey'

*Description* : *Survey* est la structure de données principale. Les topographies peuvent être imbriquées, ce qui permet de construire une structure hiérarchique. Habituellement, chaque niveau de la structure hiérarchique du relevé topographique représente des cavités ou des zones karstiques de niveaux supérieurs et de niveaux inférieurs, par exemple des galeries.

Chaque topographie a son propre espace de noms spécifié par son argument `<id>`. Les objets (tels que les stations topos ou les segments de calque (*scraps*); voir ci-dessous) qui appartiennent à une sous-topographie (*subsurvey*) de la topographie en cours sont référencés comme suit :

`<objet-id>@<soustopo-id>`,

ou, s'il y a plusieurs niveaux de classement :

`<objet-id>@<sousoustopo-id>.<soustopo-id>`.<sup>3</sup>

Cela signifie que les identificateurs d'objet ne doivent être uniques que dans le cadre d'une seule et même topographie. En conséquence, les noms des stations topo peuvent être identiques s'ils se trouvent dans des topographies différentes. Cela permet aux stations d'être numérotées à partir de 0 dans chaque relevé ou d'assembler deux cavités dans un même système karstique sans renommer les stations du relevé.

*Syntaxe* : `survey <id> [OPTIONS]  
... other Therion objects ...  
endsurvey [<id>]`

*Contexte* : `aucun, survey`

*Arguments* :

- `<id>` ▸ identifieur de topographie

*Options* :

- `namespace <on/off>` ▸ spécifie si *survey* crée un espace pour les noms (`on` par défaut)

---

<sup>3</sup> Note : il est impossible de faire référence à un objet rangé parmi les topographies de niveau supérieur.

- `declination <définition>` ▷ définit la déclinaison magnétique qui sera utilisée par défaut pour toutes les données des objets de cette topographie (qui peut être remplacée par les définitions de déclinaison dans les sous-niveaux). La `<définition>` peut prendre trois formes :

1. `[ ]` une chaîne vide. Cela réinitialisera la définition de la déclinaison magnétique.
2. `[<valeur> <units>]` définira une valeur unique de déclinaison magnétique (également pour les topographies non datées).
3. `[<date1> <value1> [<date2> <value2> ... ] <units>]` définira la déclinaison magnétique pour plusieurs dates différentes. Ensuite, la déclinaison de chaque visée sera définie en fonction de la date spécifiée par les données de l'objet. Si vous souhaitez définir explicitement la déclinaison pour les données de topographies non datées, utilisez '-' au lieu de la date.

Si aucune déclinaison magnétique n'est spécifiée mais qu'un système de coordonnées géodésique est défini, la déclinaison est automatiquement calculée à l'aide du modèle géomagnétique intégré.

**NB:** La déclinaison magnétique est positive lorsque le nord magnétique est à l'Est du nord vrai.

- `person-rename <ancien nom> <nouveau nom>` ▷ renomme une personne dont le nom a été modifié.
- `title <blabla>` ▷ description de l'objet.
- `entrance <nom-station>` ▷ spécifie l'entrée principale de la cavité représentée par cette topographie. S'il n'est pas spécifié et qu'il y a au moins une station marquée dans cette topo, celle-ci est également considérée comme une cavité. Cette information est utilisée pour l'exportation dans `cave-list`.

### '*centreline*'

*Description* : Données du relevé topographique (ligne de cheminement). La syntaxe est empruntée à *Survex* avec des modifications mineures. Le manuel *Survex* peut être utile en tant que référence supplémentaire pour l'utilisateur. Un terme synonyme à '*centerline*' peut être utilisé.

*Syntaxe* :

```
centreline [OPTIONS]
  date <date>
  team <personnes> [<rôles>]
  explo-date <date>
  explo-team <personnes>
  instrument <quantité liste> <description>
  calibrate <quantité liste> <zero erreur> [<échelle>]
  units <quantité liste> [<facteur>] <unités>
  sd <quantité liste> <valeur> <unités>
  grade < grade liste>
  declination <valeur> <unités>
  grid-angle <valeur> <unités>
  infer <quoi> <on/off>
  mark <type>
  flags <vises marqueurs>
  station <station> <commentaire> [<flags>]
  cs <système de coordonnées>
  fix <station> [<x> <y> <z> [<std x> <std y> <std z>]]
  equate <liste stations>
```

```

data <style> <readings order>
break
group
endgroup
walls <auto/on/off>
vthreshold <nombre> <unités>
extend <spec> [<station> [<station>]]
station-names <préfixe> <suffixe>
...
[SURVEY DATA]
...
endcentreline

```

*Contexte* : none, survey

*Options* :

- `id <ext_keyword>` ▷ id de l'objet
- `author <date> <personne>` ▷ auteur des données et date de création
- `copyright <date> <blabla>` ▷ copyright date et nom
- `title <blabla>` ▷ description de l'objet

*Command-like Options*:

- `date <date>` ▷ date de la topographie. Si plusieurs dates sont spécifiées, un intervalle de temps est créé.
- `explo-date <date>` ▷ date de la découverte. Si plusieurs dates sont spécifiées, un intervalle de temps est créé.

`team <personne> [<rôles>]` ▷ un membre de l'équipe des topographes. Le premier argument est son nom, les autres décrivent les rôles de la personne dans l'équipe (facultatif - non utilisé actuellement). Les mots-clés de rôle pris en charge sont les suivants : `station`, `length`, `tape`, `[back]compass`, `[back]bearing`, `[back]clino`, `[back]gradient`, `counter`, `depth`, `station`, `position`, `notes`, `pictures`, `pics`, `instruments (insts)`, `assistant (dog)`.

- `explo-team <personne>` ▷ un membre de l'équipe de découverte.
- `instrument <quantity list> <description>` ▷ description de l'instrument utilisé lors de la collecte des données du relevé topographique (mêmes mots-clés que pour le rôle de l'équipier)
- `infer <quoi> <on/off>` ▷ `'infer plumbs on'` indique au programme d'interpréter les gradients  $\pm 90^\circ$  comme HAUT / BAS (cela signifie qu'aucune correction n'est appliquée au clinomètre). `'infer equates on'`, indique au programme d'interpréter au cas par cas les visées ayant une longueur égale à 0 en tant que commandes équivalentes (ce qui signifie qu'aucune correction n'est appliquée au décamètre).
- `declination <value> <units>` ▷ définit la déclinaison magnétique pour les visées suivantes

$$\text{true bearing} = \text{measured bearing} + \text{declination}.$$

La déclinaison est positive lorsque le nord magnétique est à l'est du nord vrai. Si aucune déclinaison n'est spécifiée ou si la déclinaison est réinitialisée (-), une valeur de déclinaison valide est recherchée dans toutes les topographies dans lesquelles se trouve les données de cet objet. Voir les options de déclinaison magnétique de la commande `'survey'`.



- `grid-angle <value> <units>` ▷ spécifie l'angle de la grille magnétique (déclinaison par rapport au nord de la grille).
- `sd <quantity list> <valeur> <unités>` ▷ (sd mis pour « déviation standard ») définit l'écart-type pour les mesures données. La liste de quantité peut contenir les mots-clés suivants : `length`, `tape` (décamètre), `bearing`, `compass`, `gradient`, `clino`, `counter` (topofil), `depth`, `x`, `y`, `z`, `position`, `easting`, `dx`, `northing`, `dy`, `altitude`, `dz`.

Pour utiliser correctement cette commande, vous devez comprendre ce qu'est un écart-type (ou « déviation standard / SD » en anglais). Il attribue une valeur à la "propagation" des erreurs dans une mesure. En supposant que celles-ci soient normalement distribuées (courbe de Gauss), nous pouvons affirmer que 95,44% des longueurs réelles se situeront dans les limites de deux écarts-types (2SD) de la longueur mesurée. Prenons l'exemple des mesures de longueur : un écart-type de 0,25 mètre signifie que la longueur réelle de ces mesures se situe dans la limite de +/- 0,5 mètre de leur valeur enregistrée par l'instrument dans 95,44% des cas. Donc, si la mesure est de 7,34 m, la longueur réelle sera très probablement comprise entre 6,84 m et 7,84 m. NB : cet exemple correspond à la « classe 3 » de la BCRA (British Cave Research Association).

- `grade <liste de grade>` ▷ définit les écarts-types en fonction de la classification spécifique des topographies établie par la BCRA = British Cave Research Association (voir détails plus loin pour la commande `grade`). Tous les écarts-type (`sd`) ou degrés (`grade`) spécifiés précédemment sont perdus. Si vous souhaitez changer de SD, utilisez l'option `sd` après cette commande. Si plusieurs degrés de précision différents sont spécifiés, seul le dernier s'applique. Vous pouvez spécifier des écarts-types pour toute la topographie ou uniquement pour une position donnée. Si vous souhaitez combiner les deux, vous devez les utiliser dans une seule ligne `grade`.
- `units <quantity list> [<facteur>] <unités>` ▷ définir les unités utilisées pour les mesures données (même contenu de liste de quantité que pour `sd`).
- `calibrate <quantity list> <zéro erreur> [<échelle>]` ▷ `calibrate` est utilisé pour spécifier les corrections (étalonnage) d'instruments de mesure, via une erreur de zéro et un facteur d'échelle. Par défaut, l'erreur zéro est de 0.0 et le facteur d'échelle de 1.0 pour toutes les quantités mesurées. La valeur mesurée sera ensuite calculée à l'aide de la formule suivante : valeur mesurée = (valeur lue - erreur de zéro) × échelle. Même contenu de liste de quantité que pour `sd` (voir ci-dessus).
- `break` ▷ peut être utilisé avec des données entrelacées pour séparer deux cheminements.
- `mark [<liste de stations>] <type>` ▷ définit le type de stations nommées. `<type>` est l'un des éléments suivants : 'fixed' (fixe), 'paint' (marquage à la peinture) et 'temporary' (temporaire / valeur par défaut). S'il n'y a pas de liste de stations, toutes les stations suivantes sont marquées avec le `type` choisi.
- `flags <shot flags>` ▷ définir des marqueurs pour les visées suivantes. Les marqueurs acceptés sont : `surface` (pour les mesures de surface), `duplicate` (pour des visées dupliquées), `splay` (pour les visées latérales d'habillage qui sont masquées sur les dessins et les modèles par défaut). Ceux-ci sont exclus des calculs de longueur.

Toutes les visées dont l'une des stations est marquée soit '.' soit '-' sont définies comme des visées latérales d'habillage par défaut (voir aussi la commande `data`).

Si le marqueur est défini sur `approximate`, la visée est incluse dans les calculs de longueur totale, mais est également affichée séparément dans les statistiques de la topographie. Ce marqueur doit être utilisé pour les visées incorrectes ou imprécises et qui doivent être réexaminées.

De plus, "not" (non) est autorisé avant un marqueur.

- `station <station> <commentaire> [<marqueurs>]` ▸ ajouter le commentaire de la station et ses marqueurs. Si "" est indiqué comme commentaire, il est ignoré.

Les marqueurs acceptés sont : `entrance` (entrée), `continuation` (suite), `air-draught[:winter/summer]` (courant d'air[:hiver/été]), `sink` (perte), `spring` (arrivée d'eau), `doline` (cône d'absorption), `dig` (surcreusement), `arch` (voûte), `overhang` (surplomb). De plus, "not" (non) est autorisé avant un marqueur, pour supprimer le marqueur ajouté précédemment.

Vous pouvez également spécifier des attributs personnalisés pour une station à l'aide du marqueur `attr`, suivi du nom et de la valeur de l'attribut.

Exemple :

```
station 4 "puits à explorer" continuation attr code "V"
```

S'il existe un secteur qui a été exploré, mais pas encore topographié, la longueur explorée estimée de ce secteur peut être ajoutée à une station avec le marqueur `continuation`. Ajoutez simplement `explored<explored-length>` aux marqueurs de la station. Les longueurs explorées font partie des statistiques topographiques de la cavité et affichées séparément.

Exemple :

```
station 40 "horrible étroiture" continuation explored 100m
```

- `cs <système de coordonnées>` ▸ système de coordonnées pour les stations ayant des coordonnées fixes.
- `fix <station> [<x> <y> <z> [<std x> <std y> <std z>]]` ▸ fixe les coordonnées de la station topo (avec des erreurs spécifiques). Seule la conversion d'unités, et non l'étalonnage, pourra leur être appliquée.
- `equate <liste des stations>` ▸ définit des points topos équivalents (égaux).
- `data <style> <ordre de lecture>` ▸ définit le style des données : `normal`, `topofil`, `diving` (plongée), `cartésien`, `cylpolaire` (coordonnées polaires cylindriques : dans ce cas les mesures de longueurs sont horizontales et non en suivant la pente), `dimensions`, `nosurvey` (relevé topo vide) et l'ordre de lecture. La zone de l'ordre de lecture utilise l'un des mots-clés suivants :

```
station, from, to, tape/length [back]compass/[back]bearin,  
[back]clino/[back]gradient, depth, fromdepth, todepth, depthchange, counter,  
fromcount, tocount, northing, easting, altitude, up/ceiling4, down/floor,  
left, right, ignore, ignoreall.
```

C'est-à-dire : station, depuis, vers, décamètre/longueur, compas/azimut et mesure inverse, clinomètre/pente et mesure inverse, profondeur, profondeur depuis, profondeur vers, changement de profondeur, compteur (topofil), compteur depuis, compteur vers, vers le nord, vers l'est, altitude, haut/voûte, bas/sol, gauche, droite, ignorer, tout ignorer.

Voir le manuel du logiciel *Survex* pour plus de détails.

Pour les données entrelacées, les mots-clés de nouvelle ligne et de direction sont pris en charge. Si des

---

<sup>4</sup> La dimension peut être spécifiée comme une paire [`<from>` `<to>`], c'est à dire la dimension au début et à la fin de la visée.

mesures avant-arrière (normale / inverse) de compas ou de clinomètre sont enregistrées, la moyenne des deux sera calculée.

Si l'une des stations de visée porte le nom '.' ou '-', le marqueur `splay` (visée latérale d'habillage) lui est attribué. Le point '.' devrait être utilisé pour les visées se terminant sur des éléments situés à l'intérieur de la galerie topographiée, le tiret '-' pour les visées se terminant sur les parois, au sol ou au plafond. Bien que *Therion* ne fasse pas encore de distinction entre eux, il devrait être utilisé pour améliorer la modélisation 3D à l'avenir.

- `group`
- `endgroup` ▷ `group/endgroup` Cette paire permet à l'utilisateur d'apporter des modifications temporaires dans presque tous les paramètres. (`calibrate`, `units`, `sd`, `data`, `flags`...)
- `walls <auto/on/off>` ▷ Active / désactive la création de la forme de la galerie (parois automatiques) à partir des données LRUD pour les visées suivantes. Si cette option est définie sur `auto`, le secteur n'est créé que s'il n'y a pas de segment de calque (`scrap`) référençant la ligne de cheminement.
- `vthreshold <nombre> <unités>` ▷ seuil pour interpréter les lectures LRUD comme des lectures gauche-droite-avant-arrière perpendiculaires à la visée (habillage).

Si le secteur est horizontal (`inclination < vthreshold`), LR est perpendiculaire à la visée et UD est vertical.

Si le secteur est plus ou moins vertical (`inclination > vthreshold`), même UD devient perpendiculaire au plan - sinon, le rendu ne serait pas correct. Dans le cas de visées verticales, la valeur UD est interprétée comme une dimension nord-sud de la station afin de permettre la modélisation tubulaire des verticales (visualisation des puits).

- `extend <précision> [<station> [<station>]]` ▷ contrôle la direction de la ligne de cheminement. `<précision>` est l'un des mots-clés suivants :

`normal/reverse` ▷ étendre les stations données et suivantes dans la même direction / ou la direction inverse par rapport à la station précédente. Si deux stations sont indiquées, la direction est appliquée uniquement à la visée donnée.

`left/right` ▷ comme ci-dessus, mais la direction est spécifiée explicitement.

`vertical` ▷ ne déplacez pas la station (visée) dans la direction X, utilisez uniquement la composante Z (verticale) de la visée.

`start` ▷ spécifie la station de départ (visée).

`ignore` ▷ ignore la station spécifiée (visée), continuer et prolonger le cheminement avec une autre station (visée) si c'est possible.

`hide` ▷ masquer (ne pas afficher) la station spécifiée (visée) en coupe développée. Si aucune station n'est spécifiée, `<spec>` est valide pour les visées spécifiées suivantes.

- `station-names <préfixe> <suffixe>` ▷ ajoute un préfixe / suffixe donné à toutes les stations topo de la ligne de cheminement en cours. Enregistre bon nombre de signes typographiques.

**'scrap'**

*Description* : Un `scrap` (segment) est un morceau de topographie 2D qui ne contient pas de parties qui se superposent (toutes les galeries pouvant être dessinées sur un papier sans se chevaucher). Pour les grottes courtes et simples, la cavité entière peut appartenir à un seul segment. Dans les systèmes complexes, un segment représente généralement une salle ou une galerie. Idéalement, un segment contient environ 100 m de cavité<sup>5</sup>. Chaque segment est traité séparément par METAPOST; les segments trop volumineux peuvent dépasser la mémoire de METAPOST et provoquer des erreurs.

Un segment de topo (`scrap`) se compose de symboles, de points, de lignes et de zones. Voir le chapitre « *Comment le dessin est-il construit ?* » pour une explication sur comment et dans quel ordre ces éléments sont affichés.

La bordure d'un segment de calque (`scrap`) est constituée de lignes avec les options `-outline out` ou `-outline in` (les parois de galerie ont `-outline out` par défaut). Ces lignes ne doivent pas se croiser - sinon Tsegmentherion (METAPOST) ne peut pas déterminer l'intérieur du segment et METAPOST renvoie un message d'avertissement "`scrap outline intersects itself`". (Le contour du segment se coupe lui-même).

Chaque bloc a son propre système de coordonnées cartésiennes local, qui correspond généralement au quadrillage sur papier millimétré (si vous mesurez manuellement les coordonnées des symboles de la topographie) ou aux pixels de l'image numérisée (si vous utilisez XTherion). Therion effectue la transformation de ce système de coordonnées local en coordonnées réelles à l'aide des positions des stations de relevé, qui sont spécifiées à la fois dans le `scrap` en tant que symboles de points topo et dans les données de la ligne de cheminement. Si le segment ne contient pas au moins deux stations de levé avec la référence `-name`, vous devez utiliser l'option `-scale` pour calibrer le segment (ceci est courant pour les sections transverses).

La transformation comprend les étapes suivantes :

- Transformation linéaire (décalage, mise à l'échelle et rotation) qui fait correspondre « au mieux » les stations dessinées du segment aux stations réelles. "Au mieux" signifie ici que la somme des distances au carré entre les stations correspondantes avant et après la transformation est minimale. Le résultat est affiché en rouge si l'option `debug` (débogage) de la commande `layout` (présentation / mise en page) est activée (`on`).
- Transformation non linéaire du segment qui (1°) déplace les stations topo à la position correcte, (2°) est continue. Résultat affiché en bleu en mode `debug` (débogage).
- Transformation non linéaire du segment qui (1°) déplace les points joints, (2°) ne déplace pas les stations topo, (3°) est continue. Enfin, la position des points de contrôle des courbes est ajustée pour préserver la douceur du rendu. Le résultat produit la topographie finale.

*Syntaxe* : `scrap <id> [OPTIONS]`

```
... commandes points, lignes et surfaces ...  
endscrap [<id>]
```

*Contexte* : aucun, `survey`

*Arguments* :

---

<sup>5</sup> Si nécessaire, les segments peuvent être beaucoup plus petits - pour afficher juste quelques mètres de la cavité. Lorsque vous décidez de la taille du segment, veuillez prendre en compte les éléments suivants : L'utilisation de petits segments peut prendre plus de temps au cartographe pour optimiser les jonctions de segments. D'un autre côté, les algorithmes de déformation de topographie déforment probablement moins les segments les plus petits. L'utilisation de segments trop volumineux peut épuiser la mémoire de METAPOST si des remplissages de passages sont fréquemment utilisés, de plus l'éditeur de topos de XTherion est beaucoup moins réactif lors de l'édition de gros morceaux.

- `<id>` ▷ identifiant de segment

#### Options :

- `projection <définition>` ▷ spécifie la projection du dessin. Chaque projection est identifiée par un type et éventuellement par un index sous la forme `type[:index]`. L'index peut être n'importe quel mot-clé. Les types de projection suivants valides sont les suivants :
  1. `none` ▷ Aucune projection, utilisée pour les sections transverses ou les topographies indépendantes des données du relevé (par exemple, numérisation d'anciennes topos où aucune donnée originelle n'est disponible). Aucun index n'est autorisé pour cette projection.
  2. `plan` ▷ projection du plan de base (par défaut).
  3. `elevation` ▷ projection orthogonale (pour une coupe projetée), qui prend éventuellement la direction de la vue comme argument (par exemple `[elevation 10]` ou `[elevation 10 deg]`).
  4. `extended` ▷ étendue (pour une coupe développée).
- `scale <définition>` ▷ est utilisé pour une pré-mise à l'échelle (conversion des coordonnées de pixels en mètres) des données du segment. Si la projection du segment est nulle, il s'agit de la seule transformation effectuée avec des coordonnées. La `<définition>` peut prendre quatre formes :
  1. `<nombre>` ▷ `<nombre>` mètres par unité de dessin.
  2. `[<nombre> <unités de longueur>]` ▷ `<nombre> <unités de longueur>` par unité de dessin.
  3. `[<num1> <num2> <unités de longueur>]` ▷ `<num1>` unités de dessin correspond en réalité à `<num2> <unités de longueur>`.
  4. `[<num1> ... <num8> [<unités de longueur>]]` ▷ il s'agit du format le plus général, dans lequel vous spécifiez, dans l'ordre, les coordonnées x et y de deux points du segment et de deux points de la réalité. Vous pouvez également spécifier des unités pour les coordonnées des "points dans la réalité". Ce formulaire vous permet d'appliquer à la fois la mise à l'échelle et la rotation.
- `cs <système de coordonnées>` ▷ suppose que les coordonnées locales (calibrées) des segments sont données dans le système de coordonnées spécifié. C'est utile pour le placement absolu des esquisses importées lorsqu'aucune station topo n'est spécifiée<sup>6</sup>
- `stations <liste de noms de stations>` ▷ stations que vous souhaitez tracer sur le segment, mais qui ne sont pas utilisées pour la transformation de celui-ci. Vous n'avez pas à les spécifier (dessiner) avec la commande `point station`.
- `sketch <nom de fichier> <x> <y>` ▷ spécification de repérage bitmap d'une esquisse sous-jacente (coordonnées de son coin inférieur gauche).
- `walls <on/off/auto>` spécifie si le segment doit être utilisé dans la reconstruction de modèle 3D.
- `flip (none)/horizontal/vertical` ▷ retourne le segment après transformation.

---

<sup>6</sup> S'il y a des stations topo dans le segment, la spécification cs est donc ignorée.

- `station-names <préfixe> <suffixe>` ▷ ajoute un préfixe / suffixe donné à toutes les stations topo du segment actuel. Enregistre bon nombre de signes typographiques.
- `author <date> <personne>` ▷ auteur des données et date de création.
- `copyright <date> <blabla>` ▷ date et nom du copyright
- `title <blabla>` ▷ description de l'objet.

## 'point'

*Description* : `point` est une commande permettant de dessiner un symbole de point sur la topographie.

*Syntaxe* : `point <x> <y> <type> [OPTIONS]`

*Contexte* : `scrap`

*Arguments* :

- `<x>` et `<y>` sont les coordonnées d'un objet.
- `<type>` détermine le type d'objet. Les termes suivants sont acceptés :

Objets spéciaux : `station`<sup>7</sup>, `section`<sup>8</sup>, `dimensions`<sup>9</sup>; (idem en fr.)

Étiquettes : `altitude`<sup>10</sup>, `date`, `height`<sup>11</sup>, `label`, `passage-height`<sup>12</sup>, `remark`, `station-name`<sup>13</sup>, `date`; (altitude, date, hauteur, étiquette, hauteur de galerie, remarque, nom de station,)

Symbole de zone "pleine"<sup>14</sup> : `bedrock`, `sand`, `raft`, `clay`, `pebbles`, `debris`, `blocks`, `water`, `ice`, `guano`, `snow`; (roche mère ou en place ou substrat rocheux, sable, gour, argile, cailloux, débris, blocs, eau, glace, guano, neige)

<sup>7</sup> Station topo. Pour chaque segment (à l'exception des segments sans projection - avec option 'none'), au moins une station ayant une référence de station (option -name) doit être spécifiée.

<sup>8</sup> `section` est une ancre permettant de placer une section transverse à cet endroit. Ce symbole n'a aucune représentation visuelle. La section transverse doit être dans un segment séparé avec la projection «none» spécifiée. Vous pouvez le spécifier via l'option -scrap.

<sup>9</sup> Utilisez l'option -value pour spécifier les dimensions de la galerie au-dessus / au-dessous du plan de la ligne médiane utilisé lors de la création d'un modèle 3D.

<sup>10</sup> Étiquette d'altitude générale. Toutes les altitudes sont exportées sous la forme d'une différence par rapport à l'origine de la grille Z (0 par défaut). Pour afficher l'altitude sur la paroi de la galerie, utilisez l'option `altitude` pour n'importe quel point constituant la ligne de cette paroi.

<sup>11</sup> Hauteur des formations à l'intérieur du secteur topographié (comme un cône d'absorption, etc.); voir plus loin pour plus de détails.

<sup>12</sup> Hauteur de la galerie ou de la salle; voir plus loin pour plus de détails.

<sup>13</sup> Si aucun texte n'est spécifié, le nom de la station la plus proche est utilisé.

<sup>14</sup> Contrairement aux autres symboles ponctuels, ceux-ci sont découpés par la bordure. Voir le chapitre « Comment la topographie est-elle construite ? »

**Spéléothèmes** : `flowstone`, `moonmilk`, `stalactite`, `stalagmite`, `pillar`, `curtain`, `helictite`, `soda-straw`, `crystal`, `wall-calcite`, `popcorn`, `disk`, `gypsum`, `gypsum-flower`, `aragonite`, `cave-pearl`, `rimstone-pool`, `rimstone-dam`, `anastomosis`, `karren`, `scallop`, `flute`, `raft-cone`, `clay-tree`; (coulée de calcite, lait de lune, stalactite, stalagmite, pilier, draperie, excentrique, fistuleuse, cristal, paroi concrétionnée, choux-fleurs, disque, gypse, fleur de gypse, aragonite, pisolithe, bassin de gour, bordure de gour, anastomose, karren (rainures de dissolution karstique), coups de gouge, flûte, cône de gour, argile)

**Équipement** : `anchor`, `rope`, `fixed-ladder`, `rope-ladder`, `steps`, `bridge`, `traverse`, `camp`, `no-equipment`; (amarrage, corde, échelle fixe, échelle de corde, marches, pont, traverse, campement, pas d'équipement;)

**Fin de galerie** : `continuation`, `narrow-end`, `low-end`, `flowstone-choke`, `breakdown-choke`, `clay-choke`, `entrance`; (continuation, extrémité étroite, extrémité basse, bouchon de calcite, étroiture inaccessible, bouchon d'argile, entrée;)

**Autres** : `dig`, `archo-material`, `paleo-material`, `vegetable-debris`, `root`, `water-flow`, `spring`<sup>15</sup>, `sink`, `ice-stalactite`, `ice-stalagmite`, `ice-pillar`, `gradient`, `air-draught`<sup>16</sup>, `map-connection`<sup>17</sup>, `extra`<sup>18</sup>, `u`<sup>19</sup>. (creusement, matériel archéologique, matériel paléologique, débris végétaux, racine, écoulement d'eau, source, perte, stalactite de glace, stalagmite de glace, pilier de glace, gradient, courant d'air, connexion topo, extra, utilisateur).

*Options :*

- `subtype <mot-clé>` ▷ détermine le sous-type de l'objet. Pour chaque type donné, voici les sous-types possibles :

`station`<sup>20</sup> (c'est le sous-type décrivant une station topo) : `temporary` (temporaire / valeur par défaut), `painted` (marquage à la peinture), `natural` (élément naturel), `fixed` (fixe);

`air-draught` (courant d'air) : `winter` (hiver), `summer` (été), `undefined` (indéterminé / valeur par défaut);

Le sous-type peut également être indiqué directement dans la spécification `<type>` en utilisant les deux points superposés `'.'` comme séparateur<sup>21</sup>.

Toute spécification de sous-type peut être utilisée avec un type défini par l'utilisateur (`u`). Dans ce cas, vous devrez également définir le symbole METAPOST correspondant (voir le chapitre « Nouveaux symboles topographiques »).

- `orientation/orient <nombre>` ▷ définit l'orientation du symbole. Si rien n'est spécifié, il est orienté vers le nord.  $0 \leq \text{number} < 360$ .

<sup>15</sup> Utilisez toujours les symboles `spring` (arrivée d'eau) et `sink` (perte) avec une flèche `water-flow` pour le débit d'eau.

<sup>16</sup> Le nombre de coches est défini en fonction de l'option `-scale`.

<sup>17</sup> Point virtuel, utilisé pour indiquer la connexion entre des topos déplacées (coupe développée, décalage de topographie).

<sup>18</sup> Ajout de points au morphing.

<sup>19</sup> Pour définir des points de symboles par utilisateur.

<sup>20</sup> Si le sous-type de station n'est pas spécifié, `Therion` le lit à partir de la ligne de cheminement, s'il est spécifié à cet endroit.

<sup>21</sup> Par exemple `station:fixed`

- `align` ▷ alignement du symbole ou du texte. Les valeurs suivantes sont acceptées : `center`, `c`, `top` (haut), `t`, `bottom` (bas), `b`, `left` (gauche), `l`, `right` (droite), `r`, `top-left`, `tl`, `top-right`, `tr`, `bottom-left`, `bl`, `bottom-right`, `br` (aller à la ligne).
- `scale` ▷ échelle (taille) du symbole, qui peut être : `tiny` (xs), `small` (s), `normal` (m), `large` (l), `huge` (xl) ou une valeur numérique. 'normal' est la valeur par défaut. Les tailles augmentent d'un facteur  $\sqrt{2}$  donc si `xs`  $\equiv$  0.5, `s`  $\equiv$  0.707, `m`  $\equiv$  1.0, `l`  $\equiv$  1.414 and `xl`  $\equiv$  2.0.
- `place` <bottom/default/top> ▷ changements l'ordre d'affichage sur la topographie finale.
- `clip` <on/off> ▷ spécifie si un symbole est coupé par la bordure du segment de calque (scrap). Vous ne pouvez pas spécifier cette option pour les symboles suivants : `station`, `station-name`, `label`, `remark`, `date`, `altitude`, `height`, `passage-height`.
- `dist` <distance> ▷ valide pour des points supplémentaires, spécifie la distance jusqu'à la station la plus proche (ou à la station spécifiée à l'aide de l'option `-from`). Si non spécifié, la valeur appropriée des données LRUD est utilisée.
- `from` <station> ▷ valable pour des points supplémentaires, spécifie la station de référence.
- `visibility` <on/off> ▷ montre/cache le symbole.
- `context` <point/line/area> <symbol-type> ▷ (doit être utilisé avec les options `symbol-hide` et `symbol-show` de la mise en page (layout)) le symbole sera caché/affiché conformément aux règles spécifiées par <symbol-type><sup>22</sup>.
- `id` <ext\_keyword> ▷ identifiant de symbole.

#### *Options spécifiques de type de point :*

- `name` <référence> ▷ si le type de point est `station`, cette option donne la référence à la station topo réelle.
- `extend` [previous] <station>] ▷ si le type de point est `station` et que la projection du segment de calque (scrap) est en coupe développée, vous pouvez ajuster l'extension de l'axe de la ligne de cheminement à l'aide de cette option.
- `scrap` <reference> ▷ si le type de point est `section`, il s'agit d'une référence à une section transverse du segment (scrap).
- `explored` <length> ▷ Si le type de point est `continuation` (suite), vous pouvez spécifier la longueur des zones explorées mais non encore topographiées. Cette valeur est ensuite affichée dans les statistiques `survey/cave`.
- `text` ▷ texte d'une étiquette, d'une remarque ou d'une suite de galerie. Il peut contenir les mots-clés suivants pour le formatage<sup>23</sup> :

<sup>22</sup> Exemple: si vous spécifiez `-context point air-draught` sur une étiquette indiquant la date d'observation, la commande `symbol-hide point air-draught` masquera à la fois la flèche du courant d'air et l'étiquette correspondante.

<sup>23</sup> Pour un export en SVG, seuls les mots-clés `<br>`, `<thsp>`, `<it>`, `<bf>`, `<rm>` et `<lang:XX>` sont pris en compte; tous les autres sont ignorés.



`<br>` ▷ saut de ligne.

`<center>/<center>`, `<left>`, `<right>` ▷ alignement des lignes pour les étiquettes multilignes. Ignoré s'il n'y a pas de balise `<br>`.

`<thsp>` ▷ petit espace.

`<rm>`, `<it>`, `<bf>`, `<ss>`, `<si>` ▷ commutateurs de police de caractères.

`<rtl>` and `</rtl>` ▷ marque le début et la fin d'un texte écrit de droite à gauche.

`<lang:XX>` ▷ crée une étiquette multilingue (voir le type `string` (chaîne) pour une description détaillée).

- `value` ▷ valeur de la hauteur, hauteur d'une zone, étiquette d'altitude ou dimensions à un certain point.

hauteur : selon le signe de cette valeur (positif, négatif ou sans signe), ce type de symbole représente en général la hauteur d'une cheminée, la profondeur d'un trou ou la hauteur des ressauts. La valeur numérique peut éventuellement être suivie de «?» si elle est estimée. Des unités peuvent être ajoutées (par exemple, `-value [40? ft]`).

hauteur du secteur : les quatre formes de valeur suivantes sont acceptés : `+<nombre>` (la hauteur du plafond), `-<nombre>` (la profondeur du sol vers le bas ou de la profondeur de l'eau), `<nombre>` (la distance entre le plancher et la voûte) et `[+<nombre> -<number>]` (la distance au plafond et la distance au sol).

altitude : la valeur spécifiée est la différence d'altitude par rapport à la station la plus proche. Si la valeur de l'altitude est précédée par "fix" (par exemple, `-value [fix 1300]`), cette valeur est utilisée en tant qu'altitude absolue. L'altitude peut éventuellement être suivie des unités de longueur.

dimensions : `-value [<above> <below> [<units>]]` spécifie les dimensions de la zone au-dessus et au-dessous du plan d'axe utilisé dans le modèle 3D.

## 'line'

*Description* : `Line` est une commande permettant de dessiner une ligne sur la topographie. Chaque ligne est orientée et son aspect final peut dépendre de son orientation (par exemple, le sens des coches d'une bordure de pente). La règle générale est que l'espace libre (le vide) est à gauche et la roche à droite. Exemples : le bas d'une pente, le haut d'une cheminée et l'intérieur d'une galerie se trouvent respectivement à gauche des symboles de pente, de cheminée ou de paroi.

*Syntaxe* :

```
line <type> [OPTIONS]
  [OPTIONS]
  ...
  [LINE DATA]
  ...
  [OPTIONS]
  ...
  [LINE DATA]
  ...
```

endline

Contexte : scrap

Arguments :

- `<type>` est un mot-clé qui détermine le type de ligne. Les types suivants sont acceptés :

**Galleries** : `wall`, `contour`, `slope`<sup>24</sup>, `floor-step`, `pit`, `ceiling-step`, `chimney`, `over-hang`, `ceiling-meander`, `floor-meander`, `low-ceiling`, `pit-chimney`; (paroi, contour, pente, ressaut, puits, décrochement de voûte, cheminée, surplomb, méandre au plafond, méandre au sol, plafond bas, cheminée;)

**Zones remplies** : `flowstone`, `moonmilk`, `rock-border`<sup>25</sup>, `rock-edge`<sup>26</sup>, `water-flow`, `abyss`, `entrance`, `dripline`, `fault`, `joint`, `rimstone-dam`, `rimstone-pool`, `walkway`; (plancher stalagmitique, moonmilk, bordure rocheuse, arête rocheuse, écoulement d'eau, trou, chenal de voûte, faille, joint, bordure de gourd, gourd, pont rocheux;)

**Étiquettes** : `label`; (étiquette)

**Spécial**: `border`, `arrow`, `section`<sup>27</sup>, `survey`<sup>28</sup>, `map-connection`<sup>29</sup>, `u`<sup>30</sup>.

Command-like Options :

- `subtype <keyword>` ▸ détermine le sous-type de ligne. Les sous-types suivants sont valides pour les types donnés suivants :

`wall` : `invisible`, `bedrock` (par défaut), `sand`, `clay`, `pebbles`, `debris`, `blocks`, `ice`, `underlying`, `overlying`, `unsurveyed`, `presumed`, `pit`<sup>31</sup>, `flowstone`, `moonmilk`; `border`: `visible` (par défaut), `invisible`, `temporary`, `presumed`; (invisible, substrat rocheux, sable, argile, cailloux, débris, blocs, glace, sous-jacente, sus-jacente, non topographiée, présumée, puits, plancher stalagmitique, lait de lune; `bordure` : `visible` (par défaut), `invisible`, `temporaire`, `présumée`)

`water-flow`: `permanent` (par défaut), `conjectural`, `intermittent`; (permanent, supposé, intermittent)

---

<sup>24</sup> La ligne de pente marque le bord supérieur de la zone en pente. Il est nécessaire de spécifier `l-size` sur au moins un point. La longueur et l'orientation des lignes de pente sont une moyenne des tailles `l` spécifiées et des orientations aux points les plus proches. S'il n'y a pas de spécification d'orientation, les marques de pente sont perpendiculaires à la ligne de pente.

<sup>25</sup> Contour extérieur de gros rochers. Si la ligne est fermée, elle est remplie avec la couleur d'arrière-plan.

<sup>26</sup> Arêtes intérieures de gros rochers.

<sup>27</sup> Ligne indiquant la position de la section transverse. Si les deux points de contrôle (points rouges) d'une courbe de Bézier (ligne grise) sont indiqués, la ligne de section (bleue) est dessinée vers la projection perpendiculaire (en pointillé) du premier point de contrôle et à partir de la projection (en pointillé) du second point de contrôle de cette section. Aucune courbe de section n'est affichée.



<sup>28</sup> La ligne de cheminement du levé topographique est automatiquement dessinée par *Therion*.

<sup>29</sup> Utilisé pour indiquer une connexion entre des topographies (en décalage, ou les mêmes points en coupe développée).

<sup>30</sup> Utilisé pour les symboles de lignes définis par l'utilisateur.

<sup>31</sup> Habituellement ouvert à la surface.

`survey: cave` (par défaut), `surface` (valeur par défaut si la ligne de cheminement possède un marqueur de surface).

Le sous-type peut également être mentionné directement dans la spécification `<type>` en utilisant ':' comme séparateur<sup>32</sup>.

Toute indication de sous-type peut être utilisée avec un type défini par l'utilisateur (`u`). Dans ce cas, vous devrez également définir le symbole METAPOST correspondant (voir le chapitre « Nouveaux symboles topographiques »).

- `[LINE DATA]` spécifie soit les coordonnées d'un segment de ligne `<x> <y>`, soit les coordonnées d'un arc de courbe de Bézier `<c1x> <c1y> <c2x> <c2y> <x> <y>`, où `c` indique le point de contrôle.
- `close <on/off/auto>` ▷ détermine si une ligne est fermée ou non.
- `mark <mot-clé>` ▷ est utilisé pour marquer le point sur la ligne (voir la commande `join`).
- `orientation/orient <nombre>` ▷ orientation des symboles sur la ligne. S'il n'est pas spécifié, il est perpendiculaire à la droite et orienté du côté gauche.  $0 \leq \text{number} < 360$ .
- `outline <in/out/none>` ▷ détermine si la ligne sert de limite à un segment. La valeur par défaut est 'out' pour les parois, 'none' pour les autres lignes. Utiliser `-outline in` pour de gros piliers etc.
- `reverse <on/off>` ▷ si les points sont indiqués dans l'ordre inverse.
- `size <nombre>` ▷ largeur de trait (les tailles gauche et droite prendront la moitié de cette valeur).
- `r-size <nombre>` ▷ taille de la ligne à droite.
- `l-size <nombre>` ▷ même chose mais à gauche. Utilisé pour le type `slope` (pente).
- `smooth <on/off/auto>` ▷ si la ligne doit être lissée au point donné. `Auto` est la valeur par défaut.
- `adjust <horizontal/vertical>` ▷ décale le point de la ligne à aligner horizontalement / verticalement avec le point précédent (ou le point suivant s'il n'y a pas de point précédent). Le résultat est un segment de ligne horizontale / verticale). Si tous les points de la ligne ont cette option, ils sont alignés respectivement sur la coordonnée moyenne `y` ou `x`. Cette option n'est pas autorisée dans la projection `plan`.
- `place <bottom/default/top>` ▷ changements de la disposition lors de l'affichage sur la topographie finale.
- `clip <on/off>` ▷ spécifie si un symbole est coupé par la bordure du segment.
- `visibility <on/off>` ▷ affiche / cache le symbole.
- `context <point/line/area> <symbol-type>` ▷ (à utiliser avec les options de mise en page (`layout`) `symbol-hide` et `symbol-show`). Le symbole sera masqué/affiché conformément aux règles du `<symbol-type>` spécifié.

---

<sup>32</sup> Par exemple : `border:invisible`

### Options spécifiques au type :

- `altitude <value>` ▷ ne peut être spécifié qu'avec le type paroi (`wall`). Cette option crée une étiquette d'altitude sur la paroi. Toutes les altitudes sont exportées sous la forme d'une différence par rapport à l'origine de la grille Z (0 par défaut). Si la valeur est spécifiée, elle donne la différence d'altitude du point sur la paroi par rapport à la station la plus proche. La valeur peut être préfixée par le mot-clé "fix", alors aucune station proche ne sera prise en compte; la valeur absolue donnée étant utilisée à la place. Les unités peuvent suivre la valeur. Exemples : `+4`, `[+4 m]`, `[fix 1510 m]`.
- `border <on/off>` ▷ cette option ne peut être spécifiée qu'avec le type de symbole 'slope' (pente). Il active / désactive la ligne de limite de pente.
- `direction <begin/end/both/none/point>` ▷ doit être utilisé uniquement avec le type 'section'. Il indique où placer une flèche de direction sur la ligne de section transversale. 'none' est la valeur par défaut.
- `gradient <none/center/point>` ▷ ne peut être utilisé qu'avec le type de contour et indique où placer une marque de pente sur la ligne de contour. S'il n'y a pas de spécification de pente, le comportement dépend du jeu de symboles (par exemple, pas de coche dans UIS, coche au milieu dans SKBB).
- `head <begin/end/both/none>` ▷ peut être utilisé uniquement avec le type 'arrow' (flèche) et indique où placer la pointe de la flèche. 'end' est la valeur par défaut.
- `text <blabla>` ▷ valable uniquement pour les lignes d'étiquettes.
- `height <value>` ▷ profondeur d'un puits ou de la paroi du puits ; disponible dans METAPOST comme variable numérique `ATTR__height`.

### Options :

- `id <ext_keyword>` ▷ identifiant de symbole.

### 'area'

*Description* : Une zone est spécifiée par les lignes de bordure environnantes. Elles peuvent être de n'importe quel type, mais doivent être répertoriées dans l'ordre et chaque paire de lignes consécutives doit se croiser. Pour vous assurer que les lignes se coupent même après la transformation, vous pouvez, par exemple, continuer une bordure de lac 1 cm derrière une paroi de la galerie - ces chevauchements seront automatiquement découpés par une des bordures du segment. Pour y parvenir, vous pouvez utiliser une bordure invisible à l'intérieur de la galerie.

### Syntaxe :

```
area <type>
  place <bottom/default/top>
  clip <on/off>
  visibility <on/off>
  ... border line references ...
endarea
```

*Contexte* : `scrap` (segment de calque)

### Arguments :

- `<type>` c'est un des termes suivants : `water`, `sump`, `sand`, `debris`, `blocks`, `flowstone`, `moonmilk`, `snow`, `ice`, `clay`, `pebbles`, `bedrock`<sup>33</sup>, `u`<sup>34</sup>, `mudcrack`, `pillar`, `pillar-with-curtains`, `stalactite`, `stalactite-stalagmite`, `stalagmite`. (eau, puits noyé, sable, débris, blocs, coulée de calcite, moonmilk, neige, glace, argile, cailloux, roche-mère, u(tilisateur), argile fissurée, pilier, pilier avec draperies, stalactite, stalactite-stalagmite, stalagmite).

*Options Command-like :*

- Les données des lignes sont constituées de leurs références de lignes de bordure (IDs)
- `place <bottom/default/top>` ▷ changements de la disposition lors de l'affichage sur la topographie finale.
- `clip <on/off>` ▷ spécifie si un symbole est coupé par la bordure du segment (`scrap`).
- `visibility <on/off>` ▷ affiche / cache le symbole.
- `context <point/line/area> <symbol-type>` ▷ (à utiliser avec les options de mise en page (`layout`) `symbol-hide` et `symbol-show`). Le symbole sera masqué/affiché conformément aux règles du `<symbol-type>` spécifié.

*Options :*

- `id <ext_keyword>` ▷ identifiant de symbole.

## 'join'

*Description :* La jonction fonctionne selon deux modes : elle peut relier deux segments, mais aussi deux ou plusieurs points ou lignes d'une même topographie.

Lorsque vous joignez plus de deux points ou lignes, utilisez une seule commande de jonction pour chacun d'eux, et non une séquence de commandes de jonction pour des paires de points.<sup>35</sup>

Lorsque vous joignez des segments, seules les parois sont jointes. Il est préférable de placer une jonction aussi simple que possible dans la galerie, sinon vous devrez spécifier une jonction pour chaque paire d'objets à joindre<sup>36</sup>.

Lorsque vous voulez joindre plus de deux segments à la même limite de segment, une jonction manuelle doit être effectuée, les points de connexion devant être entrés dans une déclaration de jonction<sup>37</sup>.

*Syntaxe :* `join <point1> <point2> ... <pointN> [OPTIONS]`

*Contexte :* `none`, `scrap`, `survey`

<sup>33</sup> Une zone vide qui peut être utilisée pour nettoyer l'arrière-plan.

<sup>34</sup> Pour les symboles de zone définis par l'utilisateur. Ils peuvent être suivis d'un sous-type arbitraire.

<sup>35</sup> Par exemple, utiliser `join a b c`, et non `join a b` suivi de `join b c`.

<sup>36</sup> Si vous voulez qu'un objet coupé par une limite de segment continue jusqu'à un segment voisin, utilisez l'option `-clip off` pour cet objet.

<sup>37</sup> Comme par exemple `join origScrapLineWest :end upperScrapLineWest :0 lowerScrapLineWest :0` et une autre ligne de commande similaire pour les trois autres lignes de la paroi côté Est.

### Arguments :

- `<pointX>` peut être l'ID d'un symbole de point ou de ligne, éventuellement suivi d'une marque de point de ligne `<id>:<mark>` (e.g. `podangl_131@podangl:mark1`). `<mark>` peut être aussi 'end' (fin de la ligne) ou index de point de ligne (où 0 est le premier point).

Un cas particulier se présente lorsque `<point1>` et `<point2>` sont des ID de segments de calques, alors que les extrémités de ces calques les plus proches sont jointes.

### Options :

- `smooth <on/off>` indique si deux lignes doivent être connectées avec douceur.
- `count <N>` (lorsqu'il est utilisé avec des segments de calques) ▷ *Therion* essaiera de joindre des calques qui se connectent dans `N` emplacements / galeries.

### 'equate'

*Description* : Définit l'équivalence des stations topo.

*Syntaxe* : `equate <station list>`

*Contexte* : none, survey

### 'map'

*Description* : Une 'map' (carte) est un assemblage de segments ou de plusieurs topographies présentant le même type de projection. Il est possible d'inclure un relevé topographique à la `map` : cela affichera la ligne de cheminement sur le dessin. L'objet 'map' simplifie la gestion des données lors de la sélection des données à imprimer. Voir le chapitre « Comment la topographie est-elle construite ? » pour une explication plus détaillée.

(Note : `break` ne change que le niveau des cartes de scraps mais est inactif lorsqu'il est utilisé avec des atlas de cartes, car elles induisent implicitement un `break`)

*Syntaxe* : `map <id> [OPTIONS]`

```
... scrap, survey or other map references ...  
  
break  
  
... next level scrap, survey or other map references ...  
  
preview <above/below> <other map id> endmap
```

*Contexte* : none, survey

### Arguments :

- `<id>` ▷ identifiant de segment de calque

### Command-like Options :

- Les lignes de données sont constituées de références de segment de calque (scrap) ou de topographie. Notez que vous ne pouvez pas les mélanger.

- Si vous vous référez à la topo, vous pouvez spécifier le décalage auquel cette sous-topo sera affichée, ainsi que le type de prévisualisation de sa position d'origine. La syntaxe est la suivante :

```
<map reference> [<offset X> <offset Y> <units>] <above/below/none>
```

- les segments situés après une pause (*break*) seront placés à un autre niveau (ne s'applique qu'aux cartes constituées de scraps).
- `preview <above/below> <other map id>` mettra le contour de l'autre topo en tant d'aperçu par rapport à la topographie actuelle.

L'aperçu ne s'affiche que si la topographie est au niveau (`map-level`) correspondant à celui spécifié par la commande de sélection.

Utilisez la commande `revise` si vous souhaitez ajouter des topos de niveaux supérieurs à l'aperçu.

- `colo[u]r <color>` ▸ définit la couleur de la topographie; cette option annule le choix automatique lorsque la mise en page (`layout`) spécifie `colour map-fg [map]`.

#### Options :

- `projection/proj <plan/elevation/extended/none>` ▸ est requis si la topographie contient le relevé.
- `title <blabla>` ▸ description de l'objet.
- `survey <id>` ▸ associe un relevé à une topo (par exemple, toutes les statistiques de ce relevé seront utilisées lorsque cette topo sera sélectionnée en sortie).

#### 'surface'

*Description* : Spécification de surface (zone ou espace de terrain ou de sol). Il est possible de l'afficher de deux manières : sous forme de topographie numérisée (possible en topo 2D et aussi en modèle 3D<sup>38</sup>) ou sous la forme d'une grille de surface de type modèle numérique de coupe MNE (en 3D uniquement).

#### Syntaxe :

```
surface [<name>]
  cs <coordinate system>
  bitmap <filename> <calibration>
  grid-units <units>
  grid <origin x> <origin y> <x spacing> <y spacing> <x count> <y count> grid-
  flip (none)/vertical/horizontal [grid data]
endsurface
```

*Contexte* : none, survey

#### Command-like Options :

- `cs <système de coordonnées>` ▸ système de coordonnées pour le calibrage bitmap et la spécification d'origine de la grille.
- `bitmap <nom de fichier> <calibration>` ▸ topographie numérisée.

<sup>38</sup> Vous devrez saisir des données d'altitude pour afficher la carte topographique dans un modèle 3D.

`calibration` peut prendre deux formes :

1. `[X1 Y1 x1 y1 X2 Y2 x2 y2 [units]]`, où les variables majuscules X / Y sont les coordonnées sur l'image (pixels; le coin inférieur gauche valant 0 0), les variables minuscules x / y sont les coordonnées réelles. Les unités facultatives s'appliquent aux coordonnées réelles (mètres par défaut).
2. `[X1 Y1 station1 X2 Y2 station2]`, où les variables X / Y en majuscules sont les coordonnées de l'image et les `station1` et `station2` sont les noms des stations de topographie.
3. `grid-units <units>` ▶ unités dans lesquelles la grille est spécifiée.
4. `grid <origin x> <origin y> <x spacing> <y spacing> <x count> <y count>`  
`<origin x> <origin y>` ▶ spécifier les coordonnées du coin inférieur gauche (S-W) de la grille.  
`<x spacing> <y spacing>` ▶ distance entre les nœuds (intersections) de la grille dans les directions E-W et N-S.  
`<x count> <y count>` ▶ nombre de nœuds dans la ligne et nombre de lignes qui forment la grille (voir ci-dessous).
5. `[grid data]` ▶ un flux de nombres donnant l'altitude au niveau de la mer des nœuds de la grille. Il commence à l'origine de la grille et remplit la grille en lignes (depuis la ligne W à E; puis des lignes S à N).
6. `grid-flip (none)/vertical/horizontal` ▶ utile si votre grille (exportée depuis un autre programme) doit être retournée.

## 'import'

*Description* : Lecture les données du relevé topo dans différents formats (pour le moment la ligne de cheminement est traitée aux formats \* .3d, \* .plt, \* .xyz). Les stations topo peuvent être référencées dans des segments de calque (`scrap`), etc. Lors de l'importation d'un fichier *Survex* 3D, les stations sont insérées dans la hiérarchie de la topographie s'il existe une hiérarchie identique à celle du fichier 3D.

*Syntaxe* : `import <file-name> [OPTIONS]`

*Contexte* : `survey / all`<sup>39</sup>

*Options* :

- `filter <prefix>` ▶ si spécifié, seules les stations avec un préfixe donné et des visées entre elles seront importées. Le préfixe sera supprimé des noms de station.
- `surveys (create)/use/ignore` ▶ indique comment importer une structure de relevé topographique (fonctionne uniquement avec les fichiers .3d).
  - `create` ▶ diviser les stations en sous-niveaux. S'il n'y en a pas, les créer.
  - `use` ▶ répartir les stations en sous-niveaux.
  - `ignore` ▶ ne pas répartir les stations en sous-niveaux.

---

<sup>39</sup> Uniquement avec les fichiers 3D, où la structure du relevé topo est spécifiée.



- `cs <ystème de coordonnées>` ▷ système de coordonnées pour les stations avec des coordonnées fixes.
- `calibrate [<x> <y> <z> <X> <Y> <Z>]` ▷ les coordonnées du fichier importé sont transférées des coordonnées minuscules aux coordonnées majuscules.

### 'grade'

*Description* : Cette commande est utilisée pour stocker des précisions sur les données prédéfinies pour la ligne de cheminement. Les grades ou degrés intégrés sont : BCRA<sup>40</sup> et UISv1<sup>41</sup>.

Voir la description de l'option `sd` pour la commande `centerline` afin de définir vos propres degrés de précision.

*Syntaxe* :

```
grade <id>
...
[<quantity list> <value> <units>]
...
endgrade
```

*Contexte* : all

### 'revise'

*Description* : Cette commande est utilisée pour définir ou modifier les propriétés d'un objet déjà existant.

*Syntaxe* : Pour les objets créés avec des commandes «single line» la syntaxe est la suivante `revise id [-option1 value1 -option2 value2 ...]`

Pour les objets créés avec des commandes «multi-lignes», la syntaxe est la suivante :

```
revise id [-option1 value1 -option2 value2 ...]
...
optionX valueX
data
...
```

<sup>40</sup> voir <http://bcra.org.uk/surveying/>; la syntaxe est : BCRA<sub>n</sub>, où n peut être 3 ou 5. Pour info, voici le classement BCRA des topographies suivant leur précision :

Degré 1 > Esquisse de faible précision où aucune mesure n'a été faite

Degré 2 > Peut être utilisée, si nécessaire, pour décrire un croquis dont la précision est intermédiaire entre les niveaux 1 et 3 (à utiliser uniquement si nécessaire).

Degré 3 > Relevé magnétique approximatif. Angles horizontaux et verticaux mesurés à ± 2,5°; distances mesurées à ± 50 cm; erreur de position de la station inférieure à 50 cm.

Degré 4 > Peut être utilisée, si nécessaire, pour décrire une topo qui ne répond pas à toutes les exigences de la classe 5 mais est plus précise qu'une topo de classe 3. (à utiliser uniquement si nécessaire).

Degré 5 > Relevé magnétique. Angles horizontaux et verticaux mesurés à ± 1°; les distances doivent être observées et enregistrées au centimètre près et les positions des stations identifiées à moins de 10 cm.

Degré 6 > Relevé magnétique plus précis que la classe 5. C'est-à-dire avec une précision angulaire de ± 0,5°; les lectures du clinomètre doivent avoir la même précision. L'erreur de position de la station doit être inférieure à ± 2,5 cm, ce qui nécessitera l'utilisation de trépieds dans toutes les stations ou d'autres repères de station fixe.

Degré X > Topographie basée principalement sur l'utilisation d'un théodolite / station totale (tachéomètre).

<sup>41</sup> voir <http://www.uisic.uis-speleo.org/UISmappingGrades.pdf>; la syntaxe est : UISv1<sub>n</sub>, où n est 1 à 6 ou X; alors que 1 à 2 sont uniquement là à titre d'information et X nécessite des données `sd` dans `centerline`). Les degrés 2 et 4 ne doivent être utilisés que lorsque des conditions matérielles ont empêché la topographie de satisfaire à toutes les exigences requises pour le degré supérieur et qu'il est difficile de recommencer.

endrevise

Contexte : all

Arguments :

L'identifiant signifie ici identifiant de l'objet (car l'identifiant d'un objet que vous voulez réviser doit toujours être spécifié).

### Attributs personnalisés

Les objets `survey`, `centreline`, `scrap`, `point`, `line`, `area`, `map` et `surface` peuvent contenir des attributs définis par l'utilisateur sous la forme `-attr<nom><valeur>`. `<nom>` peut contenir des caractères alphanumériques, `<valeur>` est une chaîne.

Les attributs personnalisés sont utilisés dans l'exportation de la topographie en fonction du format de sortie:

- lors de l'exportation d'un fichier de formes, elles sont écrites directement dans le fichier dbf associé. Le format de fichier de formes est un format de données vectoriel géospatial utilisé pour les logiciels de système d'information géographique (SIG).
- dans Les topographies générées à l'aide de METAPOST (PDF, SVG), les attributs sont écrits dans le fichier source METAPOST sous forme de chaînes (nommées ainsi: `ATTR_<nom>`) et peuvent être évalués et utilisés par l'utilisateur dans des macros de définition de symboles.

Vous pouvez tester la présence d'une telle variable en utilisant `if known ATTR_<name>: ... fi`.

### XTherion

*XTherion* est une GUI (interface utilisateur graphique) pour *Therion*. Il aide beaucoup à la création de fichiers de données d'entrée. Actuellement, il fonctionne dans trois modes principaux : éditeur de texte, éditeur de dessin et compilateur.<sup>42</sup>

Il n'est pas indispensable au fonctionnement de *Therion* lui-même. Vous pouvez modifier les fichiers d'entrée dans votre éditeur de texte favori et exécuter *Therion* à partir de la ligne de commande. *XTherion* n'est pas la seule interface graphique pouvant être utilisée avec *Therion*. Il serait possible d'en écrire une meilleure, ce qui serait plus convivial, plus WYSIWYG, plus rapide, plus robuste et plus facile à utiliser. Des volontaires ?

Ce manuel ne décrit pas des manips familières du style «Si vous souhaitez enregistrer un fichier, allez au menu Fichier et sélectionnez Enregistrer ou appuyez sur Ctrl-s». Parcourez le menu du haut pendant une minute pour vous familiariser avec *XTherion*.

Pour chaque mode de fonctionnement, il existe un menu supplémentaire à droite ou à gauche. Les sous-menus peuvent être repliés; vous pouvez les dérouler en cliquant sur le bouton du menu. Pour la plupart des menus et des boutons, il y a une courte description (traduite) dans la ligne d'état, il n'est donc pas difficile de deviner la signification de chacun d'eux. L'ordre des sous-menus sur le côté peut être personnalisé par l'utilisateur. Cliquez avec le bouton droit sur le bouton de menu et sélectionnez dans le menu celui parmi les autres menus avec lequel il doit être permuté.

---

<sup>42</sup> Nous nous intéressons ici à la création de données. Donc seuls les deux premiers modes sont décrits dans cette section. Pour les fonctionnalités du compilateur, voir le chapitre « Traitement des données ».

## ***XTherion* — module éditeur de texte**

L'éditeur de texte de *XTherion* offre certaines fonctionnalités intéressantes qui peuvent aider à la création de fichiers textes "input file" : prise en charge du codage Unicode et possibilité d'ouvrir plusieurs fichiers<sup>43</sup>.

Pour faciliter la saisie des données, il prend en charge le formatage des données de la ligne de cheminement sous forme de tableau. Il existe un menu Table de données ([Data table](#)) pour la saisie des données. Il peut être personnalisé en fonction des données de commandes de chaque utilisateur en appuyant sur une touche 'Scan format data' lorsque le curseur se trouve sous la spécification de la donnée de commande (option 'data' dans la commande 'centreligne').

## ***XTherion* — module éditeur de dessin**

L'éditeur de topos vous permet de dessiner et d'éditer des topographies (cartes) de manière totalement interactive. Mais n'en attendez pas trop. *XTherion* n'est pas un éditeur vraiment WYSIWYG. Il affiche uniquement la position et non la forme réelle des symboles de point ou de ligne dessinés. Visuellement, il n'y a pas de différence entre une excentrique et une étiquette de texte : les deux sont rendues sous forme de simples points. Le type et les autres attributs de tout objet sont spécifiés uniquement dans les menus « Contrôle de point » et « Contrôle de ligne » (Point control & Line control).

Exercice : Trouvez deux raisons importantes pour lesquelles la topographie dessinée dans *XTherion* ne peut pas être identique à la sortie *Therion*. (Si vous répondez à cette question, vous saurez pourquoi *XTherion* ne sera jamais un véritable éditeur WYSIWYG. La paresse des auteurs n'est pas la bonne réponse.)

Commençons par décrire l'utilisation typique de l'éditeur de dessins. Tout d'abord, vous devez choisir la partie de la cavité à dessiner<sup>44</sup>.

Après avoir créé un nouveau fichier dans l'éditeur de dessin, vous pouvez charger une ou plusieurs images (esquisses numérisées du relevé topo de la cavité<sup>45</sup> en tant qu'arrière-plan du dessin. Cliquez sur le bouton Insérer dans le menu Images d'arrière-plan. Malheureusement, en raison de la limitation du langage Tcl / Tk, seules sont prises en charge les images GIF, PNM et PPM (plus PNG et JPEG si vous avez installé l'extension tkimg). De plus, *XTherion* prend en charge le format XVI (image vectorielle *XTherion*), qui affiche les informations de la ligne de cheminement et les visées LRUD sur l'arrière-plan, ainsi que les données PocketTopo exportées au format *Therion* (voir ci-dessous). Toutes les images ouvertes sont placées dans le coin supérieur gauche de la zone de travail. Déplacez-les en double-cliquant et en les faisant glisser avec le bouton droit de la souris ou avec un menu. Pour obtenir de meilleures performances sur des ordinateurs plus lents, il est possible de décharger temporairement une image inutilisée de la mémoire en décochant la case à cocher « Visibility ». Il est possible d'ouvrir un fichier existant sans charger d'images d'arrière-plan à l'aide du menu « Open XP ».<sup>46</sup>

La taille et le zoom de la zone de dessin sont ajustables dans le menu correspondant. Le réglage automatique calcule la taille optimale de la zone de travail en fonction de la taille et de la position des images d'arrière-plan qui sont chargées.

Après ces étapes de préparation, vous êtes prêt pour le dessin ou, plus précisément, pour la création d'un fichier de données cartographiques qui donnera un dessin. Il est important de noter que vous créez en fait un fichier texte conforme à la syntaxe décrite dans le chapitre « Format des données ». En réalité, seul un sous-ensemble des commandes *Therion* est utilisé dans l'éditeur de dessins : commandes multi-lignes

---

<sup>43</sup> Le type d'encodage du fichier est spécifié sur la première ligne de celui-ci. Cette ligne est cachée par *XTherion* et peut être accessible indirectement à l'aide du menu de droite.

<sup>44</sup> Il est possible de dessiner plus d'un scrap par fichier, auquel cas tous les scraps inactifs sont affichés en jaune.

<sup>45</sup> *XTherion* ne peut pas redimensionner ni faire pivoter des images individuelles. Par conséquent, utilisez la même orientation, la même échelle et le même DPI pour toutes les images utilisées dans le même scrap.

<sup>46</sup> Remarque : *Therion* n'utilise pas d'images d'arrière-plan, à moins que vous ne les affectiez à des scraps à l'aide de l'option `-sketch`.

`scrap ... endscrap` pouvant contenir des `point`, `line` et `area`. (Cf. le chapitre « Format de données »). Cela correspond à une section de la topographie dessinée à la main, constituée de points, de lignes et de zones remplies.

La première étape consiste donc à définir le segment de calque (`scrap`) par une commande multiligne `scrap ... endscrap`. Dans le menu File Command (Commandes de fichier), cliquez sur le sous-menu Action et sélectionnez Insérer un scrap. Cela modifie le bouton Action en Insert scrap s'il avait une autre valeur. Après avoir appuyé sur ce bouton, un nouveau scrap sera inséré au début du fichier. Vous devriez voir apparaître les lignes suivantes dans la fenêtre d'aperçu au-dessus du bouton `Insérer un scrap`:

```
scrap - scrap1
endscrap
end of file
```

Cette fenêtre est un aperçu simplifié du fichier texte, qui sera enregistré par *XTherion*. Seules la commande (`scrap`, `point`, `line`, `text` - pour le texte, voir ci-dessous) et son type (pour `point` et `line`) ou son ID (pour `scrap`) sont affichés.

Le contenu complet de chaque commande est affiché dans le menu « Aperçu de la commande » (Command preview).

Pour modifier des commandes créées précédemment, il existe des menus supplémentaires, par exemple « Scrap control » pour le `scrap`. Ici, vous pouvez modifier l'ID (très important!) ou d'autres options. Pour plus de détails, voir le chapitre « Format des données ».

Il est maintenant possible d'insérer des symboles de points. Comme pour l'insertion d'une note, accédez au menu des commandes de fichier, cliquez sur le sous-menu Action et sélectionnez Insérer un point. Appuyez sur le bouton Insérer un point récemment renommé. Le raccourci clavier Ctrl-p permet aussi de faire la même chose. Cliquez ensuite sur l'emplacement souhaité dans la zone de travail et vous verrez un point bleu représentant un symbole de point. Ses attributs peuvent être ajustés dans le menu de contrôle de point (Point control). Vous resterez en mode "insertion" : chaque clic sur la zone de travail ajoute un nouveau symbole de point. Veillez à ne pas cliquer deux fois au même endroit - vous inséreriez alors deux symboles de point au même endroit ! Pour passer du mode "insérer" au mode "sélectionner", appuyez sur la touche '**Echap**' du clavier ou sur le bouton '**Select**' du menu Commandes de fichier.

Quel sera l'ordre des commandes dans le fichier de sortie ? Exactement le même que dans le plan du menu Commandes de Fichier. Les objets point, ligne et texte nouvellement créés sont ajoutés avant la ligne marquée dans le contour. Il est possible de changer l'ordre en sélectionnant une ligne et en appuyant sur les boutons 'Déplacer vers le bas', 'Déplacer vers le haut' ou 'Déplacer vers' (**Move down**, **Move up** ou **Move to**) dans le menu Commandes de fichier. De cette façon, vous pouvez également déplacer des objets entre des différents scraps.

Dessiner des lignes est similaire à dessiner dans d'autres programmes de dessin vectoriel fonctionnant avec les courbes de Bézier. (Devinez comment entrer dans le mode d'insertion de ligne, autrement qu'en d'utilisant le raccourci Ctrl-l.) Cliquez à l'endroit où le premier point doit se situer, puis faites glisser la souris en maintenant le bouton gauche enfoncé et relâchez-le à l'emplacement où devrait se trouver le premier point de contrôle. Puis cliquez ailleurs (ce point sera le deuxième point de la courbe) et faites glisser la souris (en ajustant simultanément le deuxième point de contrôle de l'arc précédent et le premier point de contrôle du suivant). Si cette explication semble trop obscure, on peut s'exercer à travailler avec certains programmes de dessin vectoriel standard possédant une documentation complète. La ligne sera terminée après avoir quitté le mode d'insertion. Le début et l'orientation de la ligne sont marqués par une petite coche orange à gauche au premier point.

Pour les symboles de ligne, il existe deux menus de contrôle : Contrôle de ligne et Contrôle de point de ligne (**Line control & Line point control**). Le premier définit les attributs associés à toute la courbe, comme son type ou son nom. La case à cocher 'inversion' est importante : *Therion* requiert des courbes orientées et il n'est pas rare de commencer à dessiner du mauvais côté. Le menu de contrôle de point de Ligne vous permet d'ajuster les attributs de n'importe quel point sélectionné sur la ligne, par exemple une

courbe lisse à cet endroit (activé par défaut) ou la présence de points de contrôle voisins (cases à cocher '<<' et '>>').

Les zones sont spécifiées par leurs lignes environnantes. Cliquez sur Insérer une zone (Inset area), puis sur les lignes entourant la zone souhaitée. Elles sont automatiquement insérées dans le champ Area et nommées (si elles ne le sont pas déjà). Une autre méthode consiste à les insérer en tant que commande `text`<sup>47</sup>, dont le contenu (entré dans le menu éditeur de texte de l'éditeur de dessin) est la commande multiligne habituelle `area ... endarea` (voir le chapitre « Format des données »).

Si vous tracez des scraps sans projection (`none`), il est nécessaire de calibrer la zone de dessin. L'échelle ne peut être définie que d'une seule manière dans *XTherion*, à l'aide des coordonnées de deux points (spécifiées à la fois dans le système de coordonnées de l'image et dans le système de coordonnées «réel»).

Après avoir sélectionné un scrap (cliquez sur son en-tête dans le menu Commandes Fichier), deux petits carrés rouges reliés par une flèche rouge apparaissent (par défaut, ils se trouvent dans les coins inférieurs de la zone de dessin). Vous devez les faire glisser vers des points dont les coordonnées sont connues, généralement des intersections de lignes de grille en mm sur le dessin numérisé. Si vous ne pouvez pas voir ces points, vous pouvez soit :

- Appuyer sur le bouton **Scale** dans le menu **Scraps** et cliquer à deux endroits différents de l'image, là où les extrémités de la flèche de calibration devraient se trouver, ou bien
- Déplacer le pointeur de la souris sur la position souhaitée, lire les coordonnées du pointeur dans la barre d'état et entrer ces coordonnées dans le 'Scraps control' à l'intérieur des champs à remplir de l'échelle des points de cette image. Après avoir saisi les paires de coordonnées X1, Y1 et X2, Y2, la flèche de calibrage sera déplacée en conséquence.

Ensuite, vous devez entrer les coordonnées réelles de ces points (X1, Y1, X2, Y2).

En mode sélection, vous pouvez sélectionner des objets line ou point existants et définir leurs attributs dans les menus correspondants, les déplacer ou les supprimer (Ctrl-d ou bouton Action dans le menu Commandes de fichier après avoir choisi Action sur Supprimer).

Il existe un menu 'Recherche et sélection' qui permet de basculer facilement d'un objet à l'autre et de visualiser ce que vous ne pouvez pas voir au premier coup d'œil sur le dessin. Par exemple, si vous entrez l'expression "station" et que vous appuyez sur Afficher tout (Show All), toutes les stations de l'image deviennent rouges.

*XTherion* ne vérifie pas la syntaxe. Il n'écrit que les objets dessinés avec leurs attributions dans un fichier texte. Toutes les erreurs sont détectées uniquement lorsque vous traitez ces fichiers avec *Therion*.

**CONSEIL :** la saisie simultanée de symboles du même type vous permet de gagner beaucoup de temps, car vous n'avez pas besoin de changer le type de symbole ni les options de remplissage pour chaque nouveau symbole. La boîte à options conserve l'ancienne valeur saisie et il suffit donc de changer quelques caractères<sup>48</sup>. Il est préférable de commencer par dessiner toutes les stations topo (n'oubliez pas de leur donner des noms en fonction des noms réels dans la commande de ligne de cheminement), afin que toutes les parois de galeries soient suivies par tous les autres symboles ponctuels, lignes et zones. Enfin, dessinez des coupes transverses.

## **Outils additionnels**

---

<sup>47</sup> ATTENTION! La commande `text` n'est pas une commande *Therion* ! Ce n'est qu'un surnom pour un bloc d'un texte arbitraire dans *XTherion*. Dans le fichier enregistré par *XTherion*, il n'y aura que ce que vous avez tapé dans l'éditeur de texte ou que vous voyez dans l'aperçu de commande (preview). Il peut s'agir d'une définition de zone (area) ou de tout autre choix, par exemple un commentaire commençant par un caractère "#".

<sup>48</sup> Dans le cas des stations topo, *XTherion* incrémente automatiquement le numéro de station à chaque nouveau symbole inséré.

[Help/Calibrate](#) bitmap génère un fichier MAP compatible 'OziExplorer' à partir des données de géoréférencement contenues dans un fichier topographique PDF<sup>49</sup>.

Si la topographie au format PDF a été convertie en raster à l'aide d'un programme externe, le convertisseur [Convert](#)<sup>50</sup> utilise une image raster et une topo PDF portant le même nom de base et situés dans le même répertoire pour calculer les données d'étalonnage.

Si le fichier PDF est utilisé directement, vous devez définir la résolution en DPI et le format de sortie avant la conversion automatique<sup>50</sup> au format raster.

Les données PocketTopo exportées au format *Therion*<sup>51</sup> à partir de l'application PocketTopo peuvent être importées aussi bien dans un éditeur de texte que dans un éditeur de dessin (Fichier → Importer → Exportation PocketTopo *Therion* et Images d'arrière-plan → Insérer → Exportation PocketTopo *Therion*). Le même fichier est utilisé pour les deux importations. L'importation d'une esquisse ne crée pas directement de données de segments ([scrap](#)). Le dessin est simplement affiché sur le fond d'écran comme des bitmaps numérisés et doit être numérisé manuellement.

### **Raccourcis clavier et souris dans l'éditeur de dessins**

#### *Général :*

- Ctrl+Z ▷ défaire
- Ctrl+Y ▷ refaire
- F9 ▷ compile le projet en cours
- pour sélectionner un objet dans la liste à l'aide du clavier : utilisez la touche 'Tab' dans la liste souhaitée; déplacez le curseur souligné sur l'objet souhaité; appuyer sur la barre 'espace'.
- PageUp/PageDown ▷ faire défiler vers le haut / bas dans le panneau latéral.
- Shift+PageUp/PageDown ▷ faire défiler vers le haut / bas la fenêtre de commande du fichier.

#### *Zone de dessin et images de fond :*

- Clic droit ▷ défilement de zone de dessin
- Double Clic droit sur une image ▷ déplacer l'image

#### Insérer un segment de calque (scrap) :

- Ctrl + R ▷ insert scrap

#### *Insérer une ligne :*

- Ctrl + L ▷ insérer une nouvelle ligne et entrer dans le mode 'insérer un point de ligne'

---

<sup>49</sup> Les informations d'étalonnage pour neuf points distincts sont présentes si la ligne de cheminement contient des stations fixées à l'aide d'un système de coordonnées géodésiques.

<sup>50</sup> [ghostscript](#) et [convert](#) doivent être installés sur votre système. Notez que l'installation de Windows n'inclut pas [ghostscript](#).

<sup>51</sup> C'est un format de texte spécial qui doit être importé à l'aide de *XTherion* et ne peut pas être traité directement par *Therion*.

- Clic gauche ▷ insérer un point de ligne (sans points de contrôle)
- Ctrl + Clic gauche ▷ insérer un point de ligne très proche du point existant (normalement inséré juste au-dessus du point existant le plus proche)
- Clic gauche + glisser ▷ insérer un point de ligne (avec des points de contrôle)
- Ctrl enfoncée tout en faisant glisser ▷ fixer la distance du point de contrôle précédent
- Clic gauche + glisser sur le point de contrôle ▷ déplacer sa position
- Clic droit sur un des points précédents ▷ sélectionner le point précédent en mode insertion (utile si vous souhaitez également modifier la direction du point de contrôle précédent).
- Esc ou Clic gauche sur le dernier point ▷ terminer l'insertion de ligne
- Clic gauche sur le premier point de la ligne ▷ fermer la ligne et insérer la dernière ligne

#### *Édition de ligne :*

- Clic gauche + glisser ▷ déplacer un point de ligne
- Clic gauche sur le point de contrôle + glisser ▷ déplacer le point de la ligne près du point existant (normalement, il est déplacé juste au-dessus du point existant le plus proche)

#### *Ajout d'un point de ligne :*

- sélectionnez le point avant lequel vous souhaitez insérer des points; insérer les points requis; appuyez sur Echap ou faites un clic gauche sur le point que vous avez sélectionné au début.

#### *Supprimer un point de ligne :*

- sélectionnez le point que vous souhaitez supprimer; appuyez sur Modifier ligne → Supprimer un point dans le panneau de configuration Ligne.

#### *Fractionnement de ligne :*

- sélectionnez le point où vous souhaitez fractionner la ligne; appuyez sur Modifier ligne → Séparer ligne dans le panneau de commande Ligne

#### *Insertion de point :*

- Ctrl + P ▷ permet de passer en mode «insertion de point».
- Clic gauche ▷ insérer un point à une position donnée.
- Ctrl + Clic gauche ▷ insérer un point très proche du point existant (normalement, il sera inséré juste au-dessus du point le plus proche).
- Esc ▷ sortir du mode 'point d'insertion'.

#### *Édition de point :*

- Clic gauche + glisser (drag & drop) ▷ déplacer le point

- Ctrl + Clic gauche + glisser (drag & drop) ▷ déplacer le point près du point existant (normalement, il est déplacé juste au-dessus du point existant le plus proche).
- Cliquez avec le bouton gauche de la souris et faites glisser les flèches de point pour modifier l'orientation ou la taille des points (en fonction des commutateurs donnés dans le panneau de configuration Point).

#### *Insertion de zone :*

- Appuyez sur les touches Ctrl + A ou les commandes de fichier → Insérer → zone pour passer au mode «Insérer zone».
- Faites un clic droit sur les lignes qui entourent la zone souhaitée
- Échap pour terminer l'insertion de lignes de bordure

#### *Édition de zone :*

- sélectionner la zone que vous souhaitez modifier
- pressez 'Insert' dans le champ Area pour insérer d'autres lignes à la position actuelle du curseur
- pressez 'Insert ID' pour insérer une bordure avec un ID donné à la position actuelle du curseur
- appuyez sur 'Supprimer' pour supprimer la limite sélectionnée

#### *Sélection d'un objet existant :*

- Clic gauche ▷ sélection d'un objet en haut
- Faites un clic droit ▷ sélectionnez un objet juste en dessous de l'objet supérieur (utile lorsque plusieurs points se superposent)

## **Penser Therion**

Bien que tout (ou presque tout) concernant les fichiers d'entrée de *Therion* ait été expliqué, ce chapitre propose quelques astuces et conseils supplémentaires.

### **Comment entrer une ligne de cheminement ?**

Le bloc de construction de base est la commande `centreline`. Si le développement de la cavité est supérieur à quelques mètres, il est judicieux de fractionner les données dans un plus grand nombre de fichiers et de séparer les données de la ligne de cheminement des données cartographiques.

Nous utilisons généralement un fichier \* .th contenant la ligne de cheminement (`centerline`) pour chaque séance de topographie. Il est pratique de commencer avec un fichier modèle vide, comme indiqué ci-dessous, où les petits points seront ensuite remplacés par les textes appropriés.

```
encoding ISO8859-1
survey ... -title "...
  centreline
    team "...
    team "...
    date ...
    units clino compass grad
    data normal from to compass clino length
    ... ..
```



```
endcentreline
endsurvey
```

Pour créer un espace de nom unique, la commande `centreline` est incluse dans la commande `survey ... end-survey`. C'est utile lorsque le relevé porte le même nom que le fichier qui le contient<sup>52</sup>. Les points seront ensuite référencés avec le caractère `@` - voir la description de la commande levée.

Pour les très grandes cavités, il est possible de construire une structure hiérarchique des répertoires. Dans un tel cas, nous créons un fichier spécial appelé `INDEX.th`, qui contient tous les autres fichiers `*.th` d'un répertoire donné et contient aussi des commandes `equate` permettant de définir les connexions entre les topographies.

### **Comment dessiner des topographies ?**

Le plus important est de concevoir la division de la cavité en plusieurs segments. Scrap est la pierre angulaire de la topographie. C'est généralement une mauvaise idée d'essayer d'ajuster chaque segment au fichier correspondant. La raison en est que les connexions entre les segments doivent être aussi simples que possible. En règle générale, les segments sont indépendants de la hiérarchie de la ligne de cheminement. Essayez donc d'oublier la hiérarchie des relevés lorsque vous tracez des topographies et choisissez les meilleures jonctions.

Nous insérons généralement des topos dans l'avant-dernier niveau de la hiérarchie du relevé<sup>53</sup>. Chaque segment de calque (scrap) peut contenir une partie arbitraire de n'importe quelle topographie du dernier niveau hiérarchique. Par exemple, une topographie principale contient les topographies `a`, `b`, `c` et `d`. Les relevés `a - d` contiennent les données de la ligne de cheminement de quatre secteurs du relevé et chacun d'eux se trouve dans un fichier séparé. Il existe une topographie `main_map` qui contient les scraps `s1` et `s2`. Si la topographie principale est située dans le relevé principal (`main`), le segment `s1` peut couvrir une partie de la ligne de cheminement de la topographie `a`, compléter le relevé `b` et une partie de `c`; `s2` couvrira une partie des topographies `a` et `c` et une topographie complète `d`. Les noms des stations du relevé seront référencés à l'aide du symbole `@` (par exemple `1@a`) dans les segments de calque (scraps)<sup>54</sup>.

Les segments sont généralement stockés dans des fichiers `*.th2`. Chaque fichier peut contenir plusieurs segments. Pour garder les données bien organisées, il existe quelques conventions de dénomination: dans le fichier `foo.th2`, tous les segments sont nommés `foo_si`, où `i` est égal à `1`, `2`, etc. Les sections transverses sont nommées `foo_ci`, les lignes `foo_li`, etc. Cela aide beaucoup avec les grands systèmes karstiques : si un segment est référencé, vous savez immédiatement dans quel fichier il a été défini.

Comme pour les fichiers `*.th`, il peut y avoir un fichier `INDEX.th2` par répertoire qui inclut tous les fichiers `*.th2`, définit les segments joints et Les topographies finales.

Lorsque vous tracez des segments, vérifiez si le contour est correctement défini : toutes les lignes créant la bordure extérieure doivent avoir l'option `outline in` ; toutes les lignes entourant les piliers intérieurs l'option `outline in`. Les contours du segment ne peuvent pas se croiser, sinon la face intérieure du segment ne peut pas être déterminée. Il y a deux tests simples pour savoir si le contour du segment est correct :

- il n'y a pas d'avertissement METAPOST "`scrap outline intersects itself`" (le contour du segment se croise)

---

<sup>52</sup> Par exemple `survey entrance` dans le fichier `entrance.th`.

<sup>53</sup> N'oubliez pas que les relevés créent des espaces de noms (qui garantissent que tous les objets d'un ensemble donné ont des noms uniques afin qu'ils puissent être facilement identifiés). Vous ne pouvez donc faire référence qu'aux objets du relevé et de tous ses sous-niveaux.

<sup>54</sup> Si vous incluez des topos dans le relevé de niveau supérieur, vous pouvez référencer n'importe quelle station dans n'importe quelle feuille, ce qui est très flexible. Mais par contre, vous devrez utiliser des noms plus longs dans les références de stations, comme `3@dno.katakombj.jmn.dumbier`

- lorsque vous définissez le remplissage de zone sur n'importe quelle couleur (`color map-fg <nombre>` option `in layout`), vous pouvez voir ce que *Therion* considère comme se trouvant à l'intérieur du segment.

### **Comment créer des modèles ?**

Un modèle est créé à partir des contours du segment. La hauteur et la profondeur de la zone sont calculées à partir des symboles de points topo `passage-height` et `dimensions`.

## ***Therion* en profondeur**

### **Comment le dessin est-il construit ?**

Ce chapitre explique comment les options `-clip`, `-place`, `-visibility` et `-context` des commandes `point`, `line` et `area` fonctionnent exactement. Il explique également les options de couleur, de transparence, de masquage et d'affichage de symboles de la commande de présentation `layout`.

Lors de l'exportation de la topographie, *Therion* doit déterminer trois attributs pour chaque symbole de point, de ligne ou de zone : visibilité (`visibility`), coupure (`clip`) et ordre. (`place`)

(1) Le symbole est visible si tout ce qui suit est vrai :

- l'option `-visibility` est activée (par défaut pour tous les symboles),
- il n'a pas été masqué par l'option `-symbol-hide` dans la présentation (`layout`),
- si son option `-context` est définie, le symbole correspondant n'a pas été masqué par l'option `-symbol-hide` de la présentation (`layout`).

Seuls les symboles visibles sont exportés.

(2) Certains symboles sont coupés par le contour du segment. Ce sont par défaut tous les éléments suivants :

- les symboles ponctuels : symboles de remplissages de zones (substrat rocheux, guano etc.),
- les symboles de ligne : tous les symboles de ligne qui n'ont pas l'option `-outline` mentionnée, à l'exception de `section`, `arrow`, `label`, `gradient` et `water-flow`.
- les symboles de surface : tous.

Le paramètre par défaut peut être modifié à l'aide de l'option `-clip`, si cela est autorisé pour un symbole particulier. Tous les autres symboles ne sont pas coupés par la limite du segment.

(3) Ordre : Chaque symbole appartient à l'un des groupes suivants, qui sont dessinés les uns après les autres :

- `bottom` ▷ tous les symboles avec l'option `-place bottom`
- `default-bottom` ▷ tous les symboles `area` par défaut
- `default` ▷ symboles qui n'appartiennent à aucun autre groupe
- `default-top` ▷ `ceiling-step` et `chimney` par défaut

- `top` ▸ tous les symboles avec l'option `-place top`

L'ordre des symboles à l'intérieur de chaque groupe suit l'ordre des commandes dans le fichier d'entrée <sup>55</sup> : les symboles qui viennent en premier sont dessinés en dernier (c'est-à-dire qu'ils sont affichés en haut de chaque groupe).

Nous sommes maintenant prêts à décrire comment les topographies (ou les chapitres de l'atlas) sont construits :

- chaque zone de la topographe finale est remplie grâce à `color map-bg`
- les surfaces bitmap sont affichées si `surface` est configurée avec `bottom`
- FOR (pour) chaque segment : le contour est rempli de blanc
- la grille est affichée si `grid` est configurée avec `bottom`
- `preview-below` <sup>56</sup> (aperçu ci-dessous) est rempli grâce à `color preview-below`
- FOR (pour) chaque niveau <sup>57</sup>:
  - BEGIN (début) de la transparence
    - FOR (pour) chaque segment de calque (`scrap`) : le contour est rempli grâce à `color map-fg`
    - FOR (pour) chaque segment de calque (`scrap`) : les symboles de zone (`area`) sont remplis et coupés (`clip`) à la limite du segment
  - END (fin) de la transparence
  - BEGIN (début) de l'écrêtage par les étiquettes de texte (pour toutes les étiquettes de ce niveau et des niveaux supérieurs)
    - FOR (pour) chaque segment de calque (`scrap`) :
      - dessiner tous les symboles à découper (à l'exception de `line survey`)
      - mettre en ordre de bas en haut
      - tracer les symboles de `line survey` qui sont coupés en bordure de segment
      - FOR (pour) chaque segment de calque (`scrap`) :
        - dessiner tous les symboles à ne pas couper (à l'exception des points de stations et de toutes les étiquettes)
        - mettre en ordre de bas en haut
        - dessiner les symboles `point station`
    - END (fin) du découpage par les étiquettes de texte
    - FOR (pour) chaque segment (`scrap`) : dessiner toutes les étiquettes (point et ligne) (y compris `wall-altitude`)
  - l'aperçu ci-dessus est dessiné avec `color preview-above`

<sup>55</sup> Ou le menu Commandes de fichiers dans *XTherion*.

<sup>56</sup> Comme spécifié à l'aide de l'option de prévisualisation (`preview`) dans la commande `map`.

<sup>57</sup> `Level` est un ensemble de scraps non séparés par un `break` dans la commande `map`.

- les bitmaps de surface sont affichés si `surface` est positionné sur `top`
- la grille est affichée si `grid` est positionné sur `top`

Ποταμοῖς τοῖς αὐτοῖς ἐμβαίνομέν τε καὶ οὐκ ἐμβαίνομεν.

*Nous marchons tous les deux et ne marchons pas dans les mêmes rivières.*  
 \* Heraclitus d' Ephesus, 6<sup>e</sup>/5<sup>e</sup> siècle Av-JC \*

## Traitement des données

Outre les fichiers de données contenant des données du relevé, *Therion* utilise un fichier de configuration contenant des instructions sur la présentation de ces données.

### Fichier de configuration

Le nom de fichier de configuration peut être donné comme argument à *Therion*. Par défaut, *Therion* recherche le fichier nommé *thconfig* dans le répertoire de travail en cours. Il se lit comme tout autre fichier *Therion* c'est-à-dire une commande par ligne; les lignes vides ou commençant par «#» sont ignorées; les lignes terminées par une barre oblique inversée (*antislash*) se poursuivant à la ligne suivante.

Une liste des commandes actuellement prises en charge est donnée ci-après :

#### 'system'

Permet d'exécuter des commandes système lors de la compilation de *Therion*<sup>58</sup>. Normalement, *Therion* attend que le sous-processus soit terminé. Si vous souhaitez continuer la compilation sans interruption, utilisez l'expression `<command> &` sous *Linux* et l'expression `<command>` sous *Windows*.

#### 'encoding'

`encoding` fonctionne comme la commande d'encodage dans les fichiers de données : il spécifie les jeux de caractères.

#### 'language'

*Syntaxe* :

- `language <xx_[YY]>`

Définit la langue de sortie pour les textes traduisibles.

#### 'cs'

*Syntaxe* :

- `cs <coordinate system>`

En dehors du bloc `layout` la commande indique le système de coordonnées de l'export. Il n'est pas possible de spécifier différents systèmes de coordonnées pour différents exports (La dernière occurrence de `cs` est utilisée pour tous les exports).

<sup>58</sup> Par exemple pour ouvrir ou actualiser le visualiseur PDF externe.

Si `cs` n'est pas défini dans le fichier de configuration, alors la première commande `cs` trouvée par *Therion* lors de la lecture des données sera utilisée pour les exports.

À l'intérieur d'un `layout` cette option indique le système de coordonnées pour la localisation des données associées (`origin`, `grid-origin`).

### '**sketch-warp**'

*Syntaxe :*

- `sketch-warp <algorithm>`

Spécifie quel algorithme de morphing utiliser pour déformer les segments de calque (scraps). Plusieurs algorithmes sont possibles comme `line` par défaut ou `plaquette` inventé by Marco Corvi.

### '**input**'

Fonctionne comme la commande `input` dans les fichiers de données. Sert à inclure d'autres fichiers.

### '**source**'

*Description :*

Spécifie les fichiers source (données) que *Therion* doit lire. Vous pouvez spécifier plusieurs fichiers ici; un par ligne. Vous pouvez également les spécifier à l'aide de l'option de ligne de commande `-s` (voir ci-dessous).

Il est également possible de taper (quelques petits extraits de code) directement dans le fichier de configuration en utilisant la syntaxe multiligne.

*Syntaxe :*

```
source <file-name>
```

ou

```
source
```

```
... commandes Therion ...
```

```
endsource
```

*Arguments :*

- `<file-name>`

### '**select**'

*Description :* sélectionne les objets (relevés topos et dessins) à exporter. Par défaut, tous les objets du relevé sont sélectionnés. Si aucune topographie n'est sélectionnée, tous les scraps (segments) provenant des topographies sélectionnées sont sélectionnés par défaut pour l'exportation de la topographie.

S'il n'y a pas de segments (scraps) ou de topos dans les données, la ligne de cheminement (`centerline`) de toutes les topographies est exportée dans la topographie.

Lorsque vous exportez des topos dans différentes projections, vous devez les sélectionner séparément pour chaque projection.

*Syntaxe* : `select <object> [OPTIONS]`

*Arguments* :

- `<object>` ▷ tout relevé ou topographie identifiés par son identifiant ID.

*Options* :

- `récursive <on / off>` ▷ valide uniquement lorsqu'une topographie est sélectionnée. Si cette option est activée (par défaut), tous les sous-niveaux du relevé donné sont sélectionnés / désélectionnés de manière récursive.
- `map-level <nombre>` ▷ valide uniquement lorsqu'une topographie est sélectionnée. Détermine le niveau auquel l'extension de la topographie pour l'exportation d'atlas est arrêtée. Par défaut, 0 est utilisé. Si «**basic**» est spécifié, l'extension se fait jusqu'aux topos de base. Remarque : les aperçus ne sont affichés que comme spécifié dans les cartes du `map-level` en cours.
- `chapter <level>` ▷ valide uniquement lorsqu'une topographie est sélectionnée. Détermine le niveau (`level`) auquel, lors de l'exportation de l'atlas, l'extension de chapitre sera stoppée. Par défaut, 0 est utilisé. Si vous spécifiez '-' ou '.', Aucun chapitre n'est exporté pour cette topographie. Si l'option de pages de titre de la mise en page est activée, chaque chapitre commence par une page de titre.

### 'unselect'

*Description* : Désélectionne les objets à exporter.

*Syntaxe* : `unselect <object> [OPTIONS]`

*Arguments* :

Identique à la commande `select`.

*Options* :

Identique à la commande `select`.

### 'maps'

*Description* : Active le traitement des cartes (par défaut) ou `off`. Si vous activez `off`, tous les scraps des topos sélectionnées seront utilisés dans la sortie, aucune définition de carte ne sera prise en compte. Utile pour déboguer les définitions de cartes.

*Syntaxe* : `maps <on/off>`

### 'maps-offset'

*Description* : Active le dessin des cartes dans `offset` (par défaut) ou `off`. Si vous activez `off`, toutes les galeries de grottes seront affichées en détail dans leur position réelle. Toutes les spécifications `offset` seront complètement ignorées.

*Syntaxe* : `maps-offset <on/off>`

### 'log'

*Description* : Activez la journalisation de diverses informations. Actuellement, seule la journalisation du traitement de la coupe développée est pris en charge.

*Syntaxe:* log extend

### **'text'**

*Description :* Spécifie la traduction de tout texte *Therion* par défaut dans le fichier de sortie.

*Syntaxe:* text <language ID> <*Therion* text> <my text>

*Arguments :*

- <language ID> ▷ identifiant de langue ISO standard (par exemple `en` ou `en_GB`)
- <*Therion* text> ▷ texte *Therion* à traduire. Pour une liste des textes *Therion* et des traductions disponibles, voir le fichier `thlang/Textes.txt`.

### **'layout'**

*Description :* Spécifie la mise en page pour les cartes en 2D. Les paramètres s'appliquant au mode Atlas sont marqués «A»; ceux pour le mode topographie par la lettre «M» (« M » mis pour 'map').

*Syntaxe:* layout <id> [OPTIONS]

```
copy <source layout id>
cs <coordinate system>
north <true/grid>
scale <picture length> <real length>
base-scale <picture length> <real length>
units <metric/imperial> rotate <number>
symbol-set <symbol-set>
symbol-assign <point/line/area/group/special> <symbol-type> \
               <symbol-set>
symbol-hide <point/line/area/group/special> <symbol-type>
symbol-show <point/line/area/group/special> <symbol-type>
symbol-colour <point/line/area/group/special> <symbol-type>
<colour> min-symbol-scale <scale>
fonts-setup <tinysize> <smallsize> <normalsize> <largesize> <hugesize>
size <width> <height> <units>
overlap <value> <units>
page-setup <dimensions> <units>
page-numbers <on/off>
exclude-pages <on/off> <list>
title-pages <on/off>
nav-factor <factor>
nav-size <x-size> <y-size>
transparency <on/off>
opacity <value>
```

```

surface <top/bottom/off>
surface-opacity <value>
sketches <on/off>
layers <on/off>
grid <off/top/bottom>
grid-origin <x> <y> <x> <units>
grid-size <x> <y> <z> <units>
grid-coords <off/border/all>
origin <x> <y> <z> <units>
origin-label <x-label> <y-label>
own-pages <number>
page-grid <on/off>
legend <on/off/all>
legend-columns <number>
legend-width <n> <units>
map-comment <blabla>
map-header <x> <y> <off/n/s/e/w/ne/nw/se/sw/center>
map-header-bg <on/off>
map-image <x> <y> <n/s/e/w/ne/nw/se/sw/center> <filename>
statistics <explo/topo/carto/copyright all/off/number>
             <explo/topo-length on/hide/off>
             <carto/copyright-count on/hide/off>
scale-bar <length> <units>
survey-level <N/all> language <xx[_YY]>
colo[u]r-model <cmyk/rgb/grayscale>
colo[u]r <item> <colour>
smooth-shading <off/quick>
debug <on/all/first/second/scrap-names/station-names/off>
doc-author <blabla>
doc-keywords <blabla>
doc-subject <blabla>
doc-title <blabla>
code <metapost/tex-map/tex-atlas>
endcode
endlayout

```

*Arguments :*

<id> ▷ identificateur de mise en page (à utiliser avec la commande `export`)

*Command-like options :*



- `copy <source layout id>` ▸ définissez ici les propriétés qui ne sont pas modifiées en fonction de la présentation source donnée (`layout`).

*Options liées à la présentation (map) de la topographie concernée :*

- `scale <picture length> <real length>` ▸ définir l'échelle de la topographie finale ou de l'atlas cartographique (M, A; par défaut : 1 200)
- `base-scale <picture length> <real length>` ▸ si défini, *Therion* mettra visuellement à l'échelle la topographie selon le facteur (`scale / base-scale`). Cela a le même effet que si la topographie imprimée à l'échelle `base-scale` était photo-réduite à l'échelle `scale`. (M, A)
- `rotate <value>` ▸ fait pivoter la topographie (M, A; par défaut: 0)
- `units <metric/imperial>` ▸ définit les unités de sortie ou d'export (M, A; par défaut: `metric`)
- `symbol-set <symbol-set>` utilise le cas échéant le jeu de symboles `symbol-set` pour tous les symboles de la topographie. Attention, le nom du jeu de symboles est sensible à la casse. (M, A)

*Therion* utilise les jeux de symboles prédéfinis suivants :

UIS (International Union of Speleology)

ASF (Australian Speleological Federation)

AUT (Austrian Speleological Association)

BCRA (British Cave Research Association)

NSS (National Speleological Society/USA)

NZSS (New Zealand Symbol Set)

SBE (Brazilian Speleological Society -Sociedade Brasileira de Espeleologia)

SKBB (Speleoklub Bansk´a Bystrica)

- `symbol-assign <point/line/area/group/special> <symbol-type> <symbol-set>` ▸ affiche un symbole particulier dans le jeu de symboles donné. Cette option annule l'option `symbol-set`.

Si le symbole possède un sous-type, l'argument `<symbol-type>` peut prendre l'une des formes suivantes : `type:subtype` ou simplement `type`, qui attribue un nouveau jeu de symboles à tous les sous-types d'un symbole donné.

Les symboles suivants ne peuvent pas être utilisés avec cette option : « point section » (qui n'est pas restitué du tout) et toutes les étiquettes de points et de lignes (étiquette, remarque, altitude, hauteur, hauteur de galerie, nom de la station, date). Voir le chapitre Modification de la disposition / Personnalisation des étiquettes de texte pour des détails sur la modification de l'aspect des étiquettes. (M, A)

Le terme `group` peut être l'un des suivants : tous, ligne de cheminement (centerline), sections, eau, spéléothèmes, galeries, glace, sédiments, équipement.

Il y a deux symboles spéciaux : la flèche indiquant le nord et la barre d'échelle.

- `symbol-hide <point/line/area/group/special> <symbol-type>` ▸ n'affiche pas de symbole particulier, ni de groupe de symboles.

Vous pouvez utiliser les commandes `group cave-centerline`, `group surface-centerline`, `point cave-station`, `point surface-station` et `group text` dans `symbol-hide` et `symbol-show`.

Utiliser `flag:<entrance/continuation/sink/spring/doline/dig>` comme un `<symbol-type>` masquer les stations avec des marqueurs particuliers (par exemple `symbol-hide point flag:entrance`).

Peut être combiné avec `symbol-show`. (M, A)

- `symbol-show <point/line/area/group/special> <symbol-type>` ▸ (M, A) afficher un symbole particulier ou un groupe de symboles. Peut être combiné avec `symbol-hide`. (M, A)
- `symbol-colou[r] <point/line/area/group/special> <symbol-type> <colour>` ▸ change la couleur d'un symbole particulier ou d'un groupe de symboles.<sup>59</sup> (M, A)
- `min-symbol-scale <scale>` ▸ définit l'échelle minimale `<scale>`, à partir de laquelle les points et les lignes sont affichés sur la carte. Par exemple, pour `min-symbol-scale M`, aucun point ni aucune ligne mis à l'échelle `S` et `XS` ne sera affiché sur la carte. `<scale>` applique le même format, en tant qu'option d'échelle, pour les points et pour les lignes.
- `fonts-setup <tinysize> <smallsize> <normalsize> <largesize> <hugesize>` ▸ spécifie la taille du texte en points. `<normalsize>` (taille normale) s'applique au point étiquette (`label`), `<smallsize>` (petite taille) s'applique à la remarque et à toutes les autres étiquettes points. Chacun d'eux peut s'appliquer à l'étiquette de ligne en fonction de son option `-size`.

Les valeurs par défaut sont 8 10 12 16 24 pour les échelles jusqu'à 1: 100; 7 8 10 14 20 les échelles jusqu'à 1: 200 ; 6 7 8 10 14 les échelles jusqu'à 1: 500 et 5 6 7 8 10 les échelles inférieures à 1:500.

*Options liées à la mise en page (layout) de la topographie concernée :*

- `size <width> <height> <units>` ▸ définir la taille de la topographie en mode Atlas. S'il n'est pas spécifié, il sera calculé à partir de `page-setup` et de `overlap`. En mode carte s'applique si `page-grid` est activée (M, A; valeur par défaut: 18 22,2 cm)
- `overlap <value> <units>` ▸ définir la taille du chevauchement en unités papier en mode Atlas ou la marge de la topographie en mode carte (M, A; par défaut : 1 cm)
- `page-setup <dimensions> <units>` ▸ définir les dimensions de la page selon l'ordre suivant : largeur du papier, hauteur du papier, largeur de la page, hauteur de la page, marge gauche et marge supérieure. Si non spécifié, elles seront calculées à partir de `size` et de `overlap` (A; par défaut : 21 29.7 20 28.7 0.5 0.5 cm)
- `page-numbers <on/off>` ▸ activer / désactiver la numérotation des pages (A; par défaut : `true`)
- `exclude-pages <on/off> <list>` ▸ exclut les pages spécifiées de l'atlas des cavités. La liste peut contenir des numéros de page séparés par une virgule ou un tiret (pour les intervalles), par ex. 2,4-7, 9, 23 signifie que les pages 2, 4, 5, 6, 7, 9 et 23 doivent être omises. Seules les pages de topographies doivent être comptabilisées. (Définissez `own-pages 0` et `title-`

---

<sup>59</sup> Remarque : le changement de couleur ne s'applique actuellement aux motifs de remplissage que si (1°) le format de sortie est PDF et (2°) la version METAPOST est au moins 1.000.

`pages off` pour obtenir les numéros de page corrects à exclure.) Les modifications de `own-pages` ou de `title-pages` n'affectent pas l'exclusion des pages. (A)

- `title-pages <on/off>` ▶ activer / désactiver l'affichage du titre des pages avant chaque chapitre de l'atlas (A; par défaut : `off`)
- `nav-factor <factor>` ▶ définir le facteur de zoom du panneau navigation de l'atlas (A; valeur par défaut : 30)
- `nav-size <x-size> <y-size>` ▶ définir le nombre de pages de l'atlas dans les deux directions du panneau de navigation (A; par défaut : 2 2)
- `transparency <on/off>` ▶ définir la transparence des galeries (les galeries sous-jacentes sont également visibles) (M, A; par défaut : `on`)
- `opacity <value>` ▶ définir la valeur d'opacité en % (option utilisée si `transparency` est sur `on`). La plage de valeurs est comprise entre 0 et 100 %. (M, A; par défaut : 70)
- `surface-opacity <value>` ▶ définit en % l'opacité de la surface bitmap (option utilisée si `transparency` est sur `on`). La plage de valeurs est comprise entre 0 et 100. (M, A; par défaut : 70)
- `surface <top/bottom/off>` ▶ définir la position du bitmap de surface au-dessus / au-dessous de la carte. (M, A; par défaut: désactivé). (M, A; par défaut : `off`)
- `sketches <on/off>` ▶ activer / désactiver l'affichage des esquisses bitmaps (morphing). (M, A; par défaut : `off`)
- `layers <on/off>` ▶ activer / désactiver les couches PDF 1.5 (M, A; par défaut : `on`)
- `grid <off/bottom/top>` ▶ activer / désactiver la grille (les valeurs des coordonnées peuvent également être affichées) (M, A; par défaut : `off`)
- `cs <coordinate system>` ▶ système de coordonnées pour `origin` et `grid-origin`
- `north <true/grid>` ▶ spécifier l'orientation par défaut de la topographie (`map`). Par défaut, le nord 'vrai' (astronomique) est utilisé. Il est ignoré lorsqu'il est utilisé avec un système de coordonnées local.
- `grid-origin <x> <y> <x> <units>` ▶ définition des coordonnées de l'origine de la grille (M, A)
- `grid-size <x> <y> <z> <units>` ▶ définir la taille de la grille en unités réelles (M, A; la valeur par défaut est égale à la taille de la barre d'échelle)
- `grid-coords <off/border/all>` ▶ spécifier à quel endroit placer les étiquettes de coordonnées sur la grille. (M, A; par défaut : `off`)
- `origin <x> <y> <z> <units>` ▶ indiquer l'origine des pages de l'atlas (M, A)
- `origin-label <x-label> <y-label>` ▶ définir l'étiquette pour la page de l'atlas dont le coin inférieur gauche correspond aux coordonnées d'origine fournies. Peut être un nombre ou un caractère. (M, A; défaut : 0 0)

- `own-pages <nombre>` ▷ définir le nombre de pages personnelles ajoutées avant la première des pages générées automatiquement en mode Atlas (requis pour une numérotation correcte des pages) (A; par défaut : 0)
- `page-grid <on/off>` ▷ afficher le plan des pages (M; par défaut : `off`)

*Options associées à la légende de la topographie (map) :*

- `map-header <x> <y> <off/n/s/e/w/ne/nw/se/sw/center>` ▷ imprimer le cartouche d'en-tête (header) de la topographie à l'emplacement spécifié par `<x>` `<y>`. L'en-tête de topo prédéfini contient des informations de base sur la cavité : nom (`name`), échelle (`scale`), flèche du nord (`north arrow`), liste des topographes (`list of surveyors`), etc. Il est entièrement personnalisable (voir le chapitre *Modifier la présentation (layout)* pour plus de détails). `<x>` est l'abscisse (gauche / droite ou E / O sur la page). `<y>` est l'ordonnée (haut / bas ou N / S sur la page). Les plages pour `<x>` et `<y>` sont comprises entre -100 et 200. Le coin inférieur gauche de la carte correspond à 0 0, le coin supérieur droit à 100 100. Le cartouche d'en-tête est aligné sur le coin ou le côté spécifié comme point d'ancrage. (M; par défaut : 0 100 nw)
- `map-header-bg <on/off>` ▷ lorsque cette option est activée, l'arrière-plan du cartouche d'en-tête de la carte est rempli avec la couleur de l'arrière-plan (par exemple pour masquer la grille de la topographie). (M; par défaut : `off`)
- `map-image <x> <y> <n/s/e/w/ne/nw/se/sw/center> <filename>` ▷ inclure l'image<sup>60</sup> spécifiée par `<filename>` sur la topographie à l'emplacement spécifié par `<x>` `<y>`. Pour les coordonnées et les détails d'alignement, voir les spécifications d'en-tête de carte (`map-header`).
- `legend-width <n> <units>` ▷ largeur de la légende (M, A; valeur par défaut: 14 cm)
- `legend <on/off/all>` ▷ afficher dans le cartouche d'en-tête la liste des symboles utilisés sur la topographie. Si l'option `all` est choisie, tous les symboles du jeu de symboles actuel seront affichés. (M, A; par défaut: `off`)
- `colo[u]r-legend <on/off>` ▷ activer / désactiver la couleur de fond `map-fg` des symboles dans la légende lorsque l'option `map-fg` est utilisée pour l'altitude, le scrap ou la topographie (`map`) (M, A)
- `legend-columns <nombre>` ▷ ajuster le nombre de colonnes de la légende (M, A; par défaut: 2)
- `map-comment <blabla>` ▷ commentaire facultatif affiché dans l'en-tête de la topographie (M)
- `statistics <explo/topo/carto/copyright all/off/number>` ou bien...
- `statistics <explo/topo-length on/hide/off>` ▷ affiche des statistiques de base; si cette option est désactivée, les membres de l'équipe sont triés par ordre alphabétique; sinon ils sont classés en fonction de leur contribution à l'exploration (longueur) et/ou au relevé topographique (M, A; par défaut : `off`).
- `scale-bar <length> <units>` ▷ définir la longueur de la barre d'échelle (M, A)

---

<sup>60</sup> Notez que vous pouvez inclure aussi un fichier PDF, ce qui peut être utile pour combiner plan et coupe développée sur un même PDF avec un très bon rendu.

- `language <xx[_YY]>` ▸ définir la langue de sortie. Les langues disponibles sont répertoriées sur la page de licence /copyright. Voir les *Annexes* si vous souhaitez ajouter ou personnaliser des traductions. (M, A).
- `colo[u]r-model <cmyk/rgb/grayscale>` ▸ sélectionner le modèle de couleur de sortie (M, A ; par défaut : CMYK). Le modèle de couleur CMYK est destiné à l'impression. La surimpression noire est utilisée pour obtenir un meilleur aspect des lignes et des textes noirs. La teinte des autres couleurs peut varier en fonction du type ou des paramètres du RIP ou du pilote d'imprimante. Le modèle de couleur RVB est destiné à la présentation sur écran ou sur projecteur multimédia. S'il est utilisé pour l'impression, le résultat dépendra des paramètres de gestion des couleurs et aussi des couleurs particulières, car toutes les couleurs RVB ne peuvent pas être représentées sur l'imprimante. Le modèle en niveaux de gris est destiné à l'impression sur des imprimantes N&B. Si les couleurs d'entrée (référence, couleur dans la mise en page) ne correspondent pas au modèle de couleur de sortie (par exemple, si seule la couleur RVB est définie dans la référence et que CMYK est sélectionné comme modèle de couleur), elles seront converties au modèle de sortie.
- `colo[u]r-profile <cmyk/rgb/grayscale> <filename>` ▸ affecter un profil ICC aux couleurs spécifiées dans le modèle de couleurs donné. Les profils ICC ne sont appliqués qu'aux dessins, et non aux images matricielles intégrées comme les croquis et les bitmaps de surface. (M, A)
- `colo[u]r <item> <colour>` ▸ personnaliser la couleur des éléments spéciaux de la topographie [`map-fg` (avant-plan), `map-bg` (arrière-plan), `preview-above` (au-dessus), `preview-below` (au-dessous), `label` (étiquette)]. La plage de couleurs est comprise entre 0 et 100 pour les niveaux de gris et sous la forme d'un triplet [0–100 0–100 0–100] pour les couleurs RGB<sup>61</sup>.

Pour `map-fg` vous pouvez utiliser `altitude`, `scrap` ou `map` comme couleurs. Dans ce cas, la topographie est colorée en fonction de l'altitude, des `scrap`s ou des `map`s.

Pour `map-bg` vous pouvez utiliser `transparent` pour supprimer complètement l'arrière-plan de la page.

Pour les étiquettes (`label`), vous pouvez activer / désactiver la couleur (`on` / `off`). Si cette option est activée, les étiquettes sont colorées à l'aide de la couleur du `scrap` associé.

- `smooth-shading <off/quick>` ▸ définit le mode d'ombrage de l'arrière-plan du `scrap`. Par défaut, les couleurs d'altitude et de profondeur sont interpolées à travers le `scrap` de manière rapide. Certains problèmes sont présents si des couleurs transparentes de symbole sont utilisées. Des modes plus précis devraient être ajoutés à l'avenir. Si désactivé, le `scrap` est rempli avec une seule couleur.
- `debug <on/all/first/second/scrap-names/station-names/off>` ▸ dessiner les `scrap`s à différentes étapes de transformation avec différentes couleurs afin de visualiser comment *Therion* modifie la topographie. Voir la description de la commande `scrap` pour plus de détails. Les points dont la distance est la plus modifiée lors de la transformation sont affichés en orange. Si `scrap-names` est spécifié, les noms des `scrap`s sont affichés pour chaque `scrap`, `station-names` affiche le nom de chaque station du relevé topo.
- `survey-level <N/all>` ▸ `N` est le nombre de niveaux de relevés topos affichés à côté du nom de la station (M, A; par défaut: 0).

*Option liées à l'export au format PDF :*

---

<sup>61</sup> Notez bien que toutes les combinaisons de couleurs ne sont pas valables ; par exemple, aucune imprimante n'imprimera CMYK [100 100 100 100]. La couverture d'encre maximale ou limite (C + M + Y + K) peut être d'environ 240 ou 300, selon l'imprimante.

- `doc-author <blabla>` ▷ définir l’auteur du document (M, A)
- `doc-keywords <blabla>` ▷ définir les mots-clés du document (M, A)
- `doc-subject <blabla>` ▷ définir le sujet du document (M, A)
- `doc-title <blabla>` ▷ définir le titre du document (M, A)

*Personnalisation :*

- `code <metapost/tex-map/tex-atlas>` ▷ Ajouter / redéfinir ici les macros T<sub>E</sub>X et METAPOST. Cela permet à l'utilisateur de configurer plusieurs choses comme les symboles définis par l'utilisateur ou la mise en page (*layout*) d'une topographie ou d'un atlas au même endroit, etc. Voir le chapitre *Modifier le layout* pour plus de détails.
- `endcode` ▷ terminer un groupe de commandes T<sub>E</sub>X et METAPOST.

**'lookup'**

*Description :* Permet de définir des cartes de consultation. Elles sont utilisées pour définir une coloration personnalisée des cartes<sup>62</sup>.

*Syntaxe :*

```
lookup <type>[:<index>] [-title "texte de titre personnalisé "]
<parameter(s)> [colour] ["texte de la légende "]
...
endlookup
```

*Arguments :*

- `<type>`▷ Il peut s'agir d'une altitude, d'une date d'explo, d'une date de topo, d'une carte ou d'un scrap.
- `<index>`▷ l'index est une chaîne de caractères définie par l'utilisateur qui permet de distinguer des cartes de consultation différentes pour un même type.
- `<parameter (s)>`▷ la chose pour laquelle vous voulez définir la couleur, par exemple une altitude avec le type 'altitude'. Il peut s'agir de valeurs distinctes ou (le cas échéant) d'un intervalle de valeurs.
- `[colour]`▷ valeur en niveaux de gris, triplet RGB, quadruplet CMYK ou une combinaison de huit valeurs (RGB, niveaux de gris, CMYK). Utilisez des crochets vides ([]) si vous souhaitez que *Therion* utilise sa palette de couleurs par défaut avec une étiquette de texte.
- `[text in legend]`▷ un texte facultatif à afficher dans la légende.

Exemple 1 : Le code ci-dessous devrait générer une échelle du rouge vers le bleu pour les valeurs d'altitudes spécifiées.

```
lookup altitude -title "Altitude legend"
700 [100 0 0] "700 m a.s.l."
680
660
640
620
```

---

<sup>62</sup> Voir la commande `color`.

```
600 [0 0 100] "below 600 m"
```

```
endlookup
```

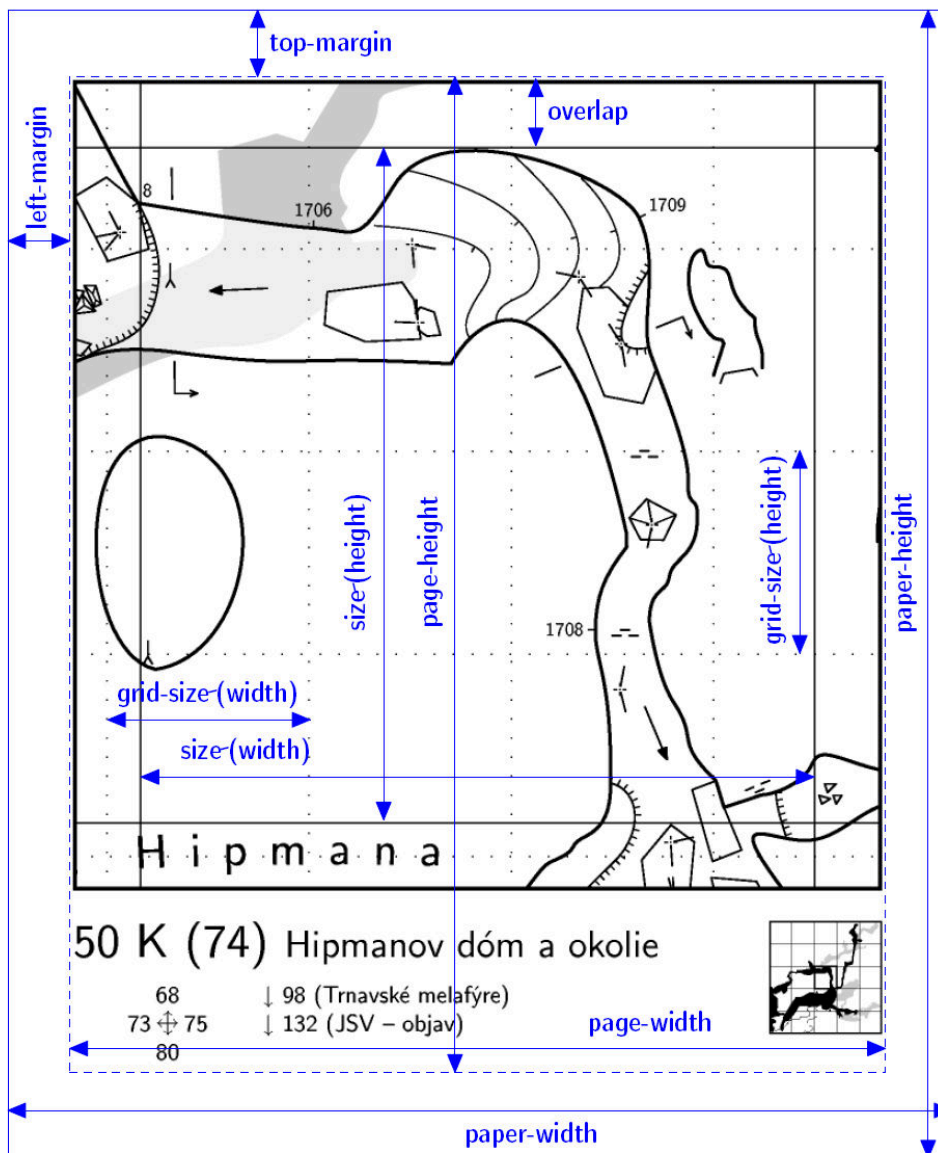
Exemple 2 : choix de l'utilisation d'un intervalle continu d'altitudes (avec index). Notez que si le texte est omis, alors la valeur par défaut affichera l'intervalle choisi, c'est-à-dire ici « 1600 m -1500 m ».

```
lookup altitude:banded
```

```
[1500 1600] [] # <-displays "1600 m -1500 m"
```

```
[1800 1900] [] "cave floor 2"
```

```
Endlookup
```



### 'setup3d'

Syntaxe :

- setup3d <value>

Hack temporaire qui permet de définir la distance d'échantillonnage (en mètres) lors de la génération d'un modèle 3D par morceaux linéaires, à partir de parois de galerie constituées de courbes de Bézier.

### 'sketch-colors'

*Syntaxe :*

- `sketch-colors <number-of-colors>`

Cette option peut être utilisée pour réduire la taille des images bitmap d'esquisses dans les tops.

## **'export'**

*Description :* Exporte les relevés ou les topographies sélectionnées.

*Syntaxe :*

- `export <type> [OPTIONS]`

*Arguments :*

- `<type>` ▸ Les types d'exportation suivants sont supportés par *Therion* :
  - `model` ▸ modèle 3D de la cavité
  - `map` ▸ topographie en 2D sur une seule page
  - `atlas` ▸ atlas topographique en 2D sur plusieurs pages
  - `cave-list` ▸ tableau de synthèse des cavités
  - `survey-list` ▸ tableau de synthèse des topographies (*survey*)
  - `continuation-list` ▸ liste des continuations possibles
  - `database` ▸ base de données SQL avec la ligne de cheminement (*centreline*)

*Options :*

*Communes :*

- `encoding/enc <encoding>` ▸ définit l'encodage de l'export
- `output/o <file>` ▸ définit le nom du fichier exporté (avec l'extension). Si aucun nom n'est spécifié, le préfixe "*cave*" est utilisé avec l'extension correspondante au format d'export.

Si le nom de fichier d'export est donné et qu'aucun format d'export n'est spécifié, le format d'export sera déterminé par l'extension du nom de fichier.

*Pour model (modèle) :*

- `format/fmt <format>` ▸ définit le format d'export. Actuellement, les formats suivants sont acceptés : `loch` (native format; default), `compass` (plt file), `Survex` (3d file), `dxg`, `esri` (3d shapefiles), `vrml`, `3dmf` and `kml` (Google Earth).
- `enable <walls/[cave/surface-]centerline/splay-shots/surface/all>` et ...
- `disable <walls/[cave/surface-]centerline/splay-shots/surface/all>` ▸ sélectionnent les caractéristiques à exporter, si le format le supporte. *Surface* n'est actuellement exporté qu'au format *Therion*.
- `wall-source <maps/centerline/all/splays>` ▸ définit la donnée source à utiliser pour la modélisation des parois 3D.



### *Pour map/atlas :*

- `format/fmt <format>` ▸ définit le format de la `map` (topographie). Actuellement sont disponibles `pdf`, `svg`, `xhtml`<sup>63</sup>, `Survex`, `dxg`, `esri`<sup>64</sup>, `kml` (Google Earth), `xvi`<sup>65</sup> et `bbox`<sup>66</sup> pour `map`; seul le format `pdf` est disponible pour l'`atlas`.
- `projection <id>` ▸ identifiant unique qui spécifie le type de projection de la `map` (topographie). (Voir la commande `scrap` pour les détails.)

Si aucune `map` n'est définie, alors ce sont les scraps de la projection donnée qui seront exportés.

S'il n'y a pas de `scrap` avec la projection spécifiée, alors `Therion` exportera la ligne de cheminement (`centerline`) des topographies (`surveys`) sélectionnées.

- `layout <id>` ▸ indique le `layout` prédéfini pour la `map` (topographie) ou l'`atlas`.
- `layout-xxx` ▸ où `xxx` permet de rajouter des options de `layout` afin de modifier certaines propriétés dans la commande export.
- `encoding/enc <encoding>` ▸ définit l'encodage de l'export.

### *Communes pour les listes (list):*

- `format/fmt <format>` ▸ définit le format d'export pour continuation. Actuellement, les formats de sorties suivants sont disponibles : `html` (par défaut), `txt`, `kml`<sup>67</sup> et `dbf`.

### *continuation-list :*

- `attributes <(on)/off>` ▸ définit si les attributs choisis par l'utilisateur dans le tableau des continuations sont exportés ou non.
- `filter <(on)/off>` ▸ définit si les continuations sans commentaires/texte doivent être filtrées ou non.

### *cave-list :*

- `location <(on)/(off)>` ▸ définit si les coordonnées des entrées doivent être exportées ou non.
- `surveys (on)/off` ▸ exporte une liste brute des cavités lorsque l'option est sur `off`. Sinon, toute la structure de la topographie (`survey`) est agrégée avec les statistiques.

### *Database :*

---

<sup>61</sup> SVG intégré au fichier XHTML qui contient aussi la légende.

<sup>62</sup> Shapefiles ESRI. Les différents fichiers du shapefile sont écrits dans un répertoire désigné par le nom de fichier spécifié.

<sup>63</sup> Image vecteur `XTherion`. Les images XVI peuvent être utilisées dans `XTherion` pour dessiner des topographies à l'échelle. `Scale` (la résolution d'image à 100 DPI est supposée) et `grid-size` du layout sont utilisés pour l'export.

<sup>64</sup> Fichier texte contenant les coordonnées géographiques des coins inférieur-gauche et supérieur-droit de la topographie.

<sup>65</sup> Pour `cave-list` et `continuation-list`.

- `format/fmt <format>` ▸ actuellement sql et csv.
- `encoding/enc <encoding>` ▸ définit l'encodage d'export.

Résumé des formats de fichiers exportés :

Type d'export	Formats disponibles
<code>model</code>	loch, dxf, esri, compass, Survex, vrml, 3dmf, kml
<code>map</code>	pdf, svg, xhtml, dxf, esri, Survex, xvi, kml, bbox
<code>atlas</code>	pdf
<code>database</code>	sql, csv
<code>lists</code>	html, txt, kml, dbf

### Faire fonctionner *Therion*

Maintenant, après avoir maîtrisé les fichiers de données et de configuration, nous sommes prêts à exécuter *Therion*. Cela se fait généralement à partir de la ligne de commande, dans le répertoire de données, en tapant

*Therion*

La syntaxe complète est :

```
therion [-q] [-L] [-l <log-file>]
        [-s <source-file>] [-p <search-path>]
        [-b/--bezier]
        [-d] [-x] [--use-extern-libs] [<cfg-file>]
```

ou

```
therion [-h/--help]
        [-v/--version]
        [--print-encodings]
        [--print-environment]
        [--print-init-file]
        [--print-library-src]
        [--print-symbols]
        [--print-tex-encodings]
        [--print-XTherion-src]
        [--reproducible-output]
        [--generate-output-crc]
        [--verify-output-crc]
```

*Arguments :*

`<cfg-file>` *Therion* ne prend qu'un seul argument optionnel : le nom du fichier de configuration. Si aucun nom de fichier de config n'est spécifié, le *thconfig* du répertoire courant est utilisé. S'il n'y a pas de fichier *thconfig* (par exemple si le répertoire en cours n'est pas le répertoire des données), *Therion* se ferme avec un message d'erreur.

*Options :*

- `-d` ▸ Force le mode débogage. L'implémentation actuelle crée un dossier temporaire nommé `thTMPDIR` (dans le répertoire temporaire de votre système) et ne supprime aucun fichier temporaire.

- `-h, --help` ▷ Afficher une aide abrégée.
- `-L` ▷ Ne pas créer de fichier journal. Normalement, *Therion* écrit tous les messages dans un fichier *Therion.log* du répertoire en cours.
- `-l <log-file>` ▷ Modifie le nom du fichier journal.
- `-p <search-path>` ▷ Cette option permet de définir le chemin de recherche (ou la liste des chemins séparés par deux points) que *Therion* utilise pour rechercher ses fichiers source (s'il ne les trouve pas dans le répertoire de travail).
- `-q` ▷ Exécute *Therion* en mode silencieux. Il n'imprimera que les messages d'avertissement et d'erreur sur STDERR.
- `--print-encodings` ▷ Imprime une liste de tous les encodages d'entrée pris en charge.
- `--print-tex-encodings` ▷ Imprime une liste de tous les encodages pris en charge pour la sortie PDF.
- `--print-init-file` ▷ Imprime un fichier d'initialisation par défaut. Pour plus de détails, voir la section d'initialisation en annexe.
- `--print-environment` ▷ Paramètres d'environnement d'impression pour *Therion*.
- `--print-symbols` ▷ Imprime une liste de tous les symboles de topographie pris en charge par *Therion* dans le fichier *symbols.xhtml*.
- `--reproducible-output` ▷ Crée des fichiers PDF et SVG reproductibles. Aucune information volatile n'est incluse (par exemple, la date de création ou la version du logiciel est omise des métadonnées). Les variables T<sub>E</sub>X `\thversion` et `\currentdate` sont fixées comme des constantes prédéfinies. Cette option impose l'utilisation de la fermeture de boucle *Therion*. Pour obtenir les mêmes fichiers de sortie sur des plateformes différentes, il est nécessaire d'installer le même jeu de polices de base utilisé par *Therion*<sup>68</sup> et d'utiliser une distribution T<sub>E</sub>X assez récente sur toutes les plateformes.
- `--generate-output-crc` ▷ Génère un file '.crc' avec une somme de contrôle CRC32 pour chaque fichier de sortie. Implique `--reproducible-output`.
- `--verify-output-crc` ▷ Afin de vérifier que le fichier de sortie n'a pas été modifié : génère une sortie reproductible, calcule la somme de contrôle CRC32 et la compare à celle enregistrée dans le fichier '.crc'
- `-s <source-file>` ▷ Définit le nom du fichier source.
- `--use-extern-libs` ▷ Pour ne pas copier les macros T<sub>E</sub>X et METAPOST dans le répertoire courant. T<sub>E</sub>X et METAPOST doivent chercher les macros où bon leur semble. À utiliser avec précaution.
- `-v, --version` ▷ Afficher les informations de version.

---

<sup>68</sup> Pour les familles de polices CM, CS et/ou CMCYR ; les fichiers de polices '.pfb' doivent être exactement les mêmes sur toutes les plateformes car ils sont intégrés dans les fichiers PDF.

- `-x` ▸ Génère le fichier '*XTherion.dat*' avec des informations supplémentaires pour *XTherion*.

## ***XTherion—compiler***

*XTherion* facilite l'exécution de *Therion*, en particulier sur les systèmes sans mode de ligne de commande. La fenêtre du compilateur est la fenêtre par défaut de *XTherion*. Pour exécuter *Therion*, il suffit d'ouvrir un fichier de configuration et d'appuyer sur la touche «F9» du clavier ou de cliquer sur le bouton «Compiler».

*XTherion* affiche les messages de *Therion* dans la partie inférieure de l'écran. Chaque message d'erreur est mis en évidence (surligné) puis lié par un lien hypertexte au fichier source où l'erreur s'est produite.

Après une première utilisation, des menus supplémentaires sont activés, comme par exemple : « Structure du relevé topo » (Survey structure) et « Structure du dessin » (Map structure). L'utilisateur peut facilement sélectionner un relevé (survey) ou une topographie (map) à exporter en double-cliquant sur certains des éléments de l'arborescence. Un simple clic dans l'arborescence de la topographie affichera des informations de base sur le relevé dans le menu « [Survey info](#) ».

## **Qu'obtenons-nous ?**

### **Les fichiers d'information**

#### ***Fichier Log***

Outre les messages de *Therion* et d'autres programmes utilisés, le fichier journal ([Therion.log](#)) contient des informations sur les valeurs calculées de la déclinaison magnétique et de la convergence des méridiens, les erreurs éventuelles, les distorsions de scrap et les transformations entre les systèmes de référence de coordonnées choisis par la bibliothèque Proj.

L'erreur de boucle absolue est la racine carrée de  $\Delta x^2 + \Delta y^2 + \Delta z^2$ , où  $\Delta x$  : c'est la différence entre les points de début et de fin (identiques) de la boucle avant la distribution d'erreur mesurée le long de l'axe x; de même pour y et z. Le pourcentage d'erreur de boucle est calculé en tant qu'erreur absolue / longueur de boucle. L'erreur moyenne est la moyenne arithmétique simple de toutes les erreurs de boucle.

La distorsion du segment de calque (scrap) est calculée à l'aide de la mesure de la distorsion définie pour toutes les paires de points (symboles de points, points et points de contrôle des symboles de ligne) dans le segment. La mesure est calculée comme  $|da - db| / db$ , où db est la distance des points avant déformation et da est la distance de points après déformation. Les distorsions maximale et moyenne du segment sont calculées en tant que maximum ou moyenne de ces mesures appliquées à toutes les paires de points.

#### ***XTherion***

*Therion* fournit des informations de base sur chaque relevé topo si l'option `-x` est activée (longueur, extensions verticale, N - S et E - W, nombre de visées et de stations). Dans *XTherion*, ces informations sont affichées dans la fenêtre de Compilation et dans le menu Info de la topographie ([survey info](#)) lorsque certaines topographies ([surveys](#)) du menu « Structure de la topographie » ([survey structure](#)) sont sélectionnées.

#### ***Export SQL***

L'exportation SQL facilite l'obtention d'informations précises et détaillées sur la ligne de cheminement ([centerline](#)). Il s'agit d'un fichier texte commençant par des déclarations de tables (dans le listing ci-dessous "?" représente la valeur maximale requise pour chaque colonne de données du tableau) :

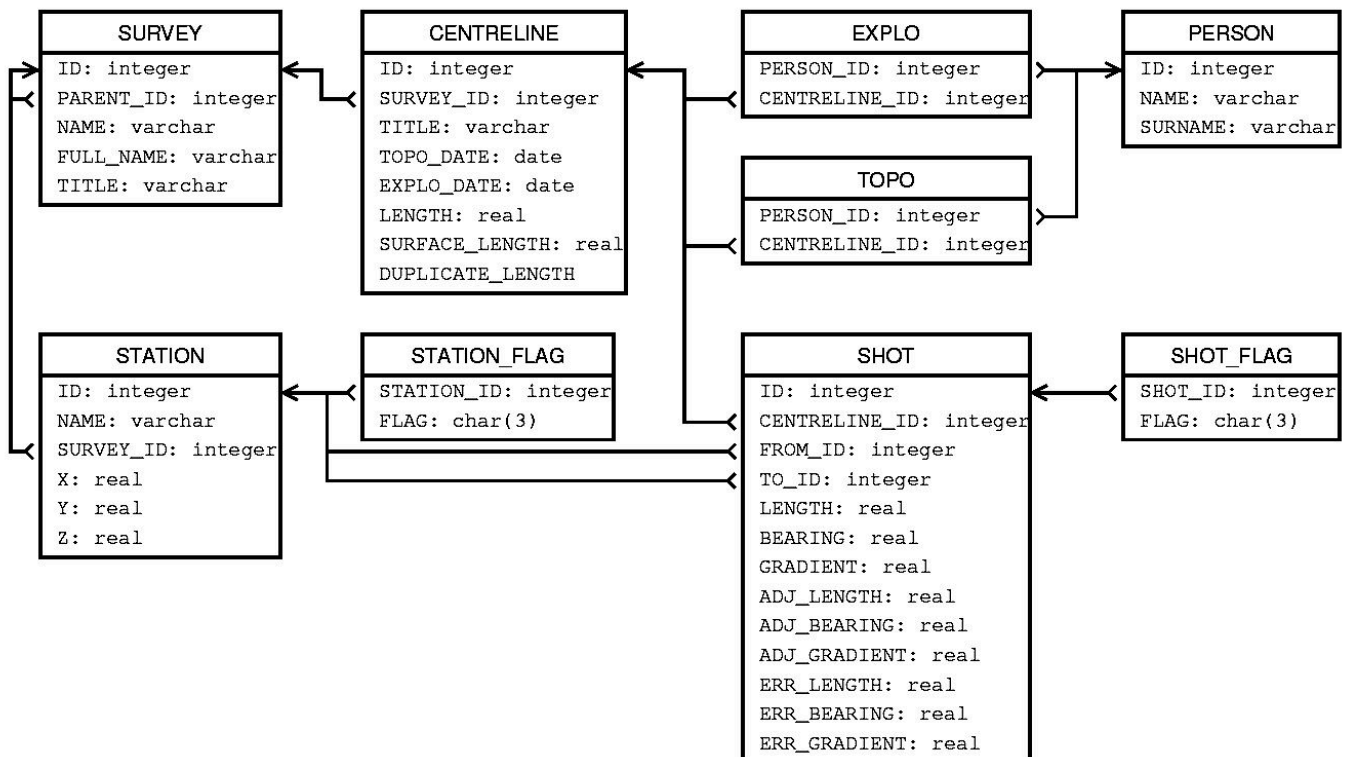
```
create table SURVEY (ID integer, PARENT_ID integer,
```

```

NAME varchar(?), FULL_NAME varchar(?), TITLE varchar(?)); create
table CENTRELINE (ID integer, SURVEY_ID integer,
TITLE varchar(?), TOPO_DATE date, EXPLD_DATE date, LENGTH real,
SURFACE_LENGTH real, DUPLICATE_LENGTH real);
create table PERSON (ID integer, NAME varchar(?), SURNAME varchar(?));
create table EXPLO (PERSON_ID integer, CENTRELINE_ID integer);
create table TOPO (PERSON_ID integer, CENTRELINE_ID integer);
create table STATION (ID integer, NAME varchar(?), > >
SURVEY_ID integer, X real, Y real, Z real);
create table STATION_FLAG (STATION_ID integer, FLAG char(3));
create table SHOT (ID integer, FROM_ID integer, TO_ID integer,
CENTRELINE_ID integer, LENGTH real, BEARING real, GRADIENT real,
ADJ_LENGTH real, ADJ_BEARING real, ADJ_GRADIENT real,
ERR_LENGTH real, ERR_BEARING real, ERR_GRADIENT real);
create table SHOT_FLAG (SHOT_ID integer, FLAG char(3));

```

Tout cela est suivi par une masse de commandes d'insertion SQL. Ce fichier peut être chargé dans n'importe quelle base de données SQL (après une initialisation dépendant de la base de données, qui peut inclure l'exécution d'un serveur SQL et la connexion à ce dernier, la création d'une base de données et la connexion à celle-ci). Il est souhaitable de démarrer une transaction informatique avant de charger ce fichier (si la base de données ne démarre pas automatiquement la transaction.) Il est important de configurer l'encodage de la base de données pour qu'il corresponde à celui spécifié dans la commande *Therion export database*.



Les noms de tableau et de colonnes s'expliquent par eux-mêmes; pour les valeurs indéfinies ou non-existantes, la valeur *NULL* est utilisée. *FLAG* dans le tableau *SHOT\_FLAG* est *dpl* ou *srf* pour les visées dupliquées ou de surface; dans le tableau *STATION\_FLAG* les abréviations *ent*, *con*, *fix*, *spr*, *sin*, *dol*, *dig*, *air*, *ove*, *arc* sont utilisées respectivement pour les stations ayant les attributs suivants : entrance (entrée), continuation, fixed (fixe), spring (source, arrivée d'eau), sink (perte), doline, dig (désobstruction), air-draught (courant d'air), overhang (surplomb) or arch (arche).

Voici quelques exemples de requêtes simples de la base de données SQL de *Therion* :

Liste des membres de l'équipe topo avec une information sur la longueur topographiée par chacun d'entre eux :

```
select sum(LENGTH), sum(SURFACE_LENGTH), NAME, SURNAME
  from CENTRELINE, TOPO, PERSON
  where CENTRELINE.ID = TOPO.CENTRELINE_ID and PERSON.ID = PERSON_ID
  group by NAME, SURNAME order by 1 desc, 4 asc;
```

Quelles parties de la cavité ont été topographiées durant l'année 1998 ?

```
select TITLE from SURVEY where ID in
  (select SURVEY_ID from CENTRELINE where TOPO_DATE between '1998-01-01' and
  '1998-12-31');
```

Quelle longueur de galerie y-a-t-il entre les altitudes 1500 et 1550 m au-dessus du niveau de la mer ?

```
select sum(LENGTH) from SHOT, STATION S1, STATION S2
  where (S1.Z+S2.Z)/2 between 1500 and 1550 and
  SHOT.FROM_ID = S1.ID and SHOT.TO_ID = S2.ID;
```

### Listes de cavités, surveys (topographies), continuations (points d'interrogation)

Utiliser l'`export continuation-list` nous permet d'obtenir un état des lieux de tous les points de la ligne de cheminement (centerline) et des scraps qui possèdent l'information d'une continuation possible<sup>69</sup>.

`export cave-list` produit sous la forme d'un tableau l'information sur les différentes cavités topographiées (surveys) (pour cela, vous devez spécifier le marqueur `entrance` dans vos données), ce qui inclut la longueur, la profondeur et la localisation des entrées.

L'information détaillée de chaque sous-topographie (sub-survey) est donnée par la commande `export survey-list`. La longueur inclut les visées avec le marqueur `approximate`, mais pas celles avec les marqueurs `explored` (exploré), `duplicate` (dupliqué) ou `surface`

## Topographies en 2D

### Topographies à imprimer

Les topographies sont produites aux formats PDF et SVG, qui peuvent être visualisés ou imprimés dans un large éventail de logiciels. Pensez bien à décocher la case *Fit page to paper* (adapter à la dimension du papier) ou toute option similaire si vous souhaitez imprimer à l'échelle exacte que vous avez définie.

En mode Atlas, des informations supplémentaires sont ajoutées à chaque page : numéro de page, nom de la topographie et étiquette de page.

Les numéros des pages voisines dans les directions N, S, E et W, ainsi que les niveaux supérieurs et inférieurs sont particulièrement utiles. Il existe également des liens hypertextes sur les côtés de la topographie si la cavité continue à la page suivante ou menant à la page appropriée correspondant à la suite de la cavité dans cette direction.

Les fichiers PDF sont hautement optimisés : les segments de calque (*scraps*) sont stockés comme XObject une seule fois dans le document, ensuite ils sont simplement référencés dans les pages appropriées. *Therion* utilise les propriétés les plus avancées des fichiers PDF telles que la transparence et les calques.

Les fichiers PDF créés peuvent éventuellement être modifiés en post-traitement dans des applications

---

<sup>69</sup> En utilisant l'attribut `station` pour un point de la ligne de cheminement (`centerline`) et `point continuation` dans les segments de calque (`scraps`).

telles que *pdf<sub>TEX</sub>* ou *Adobe Acrobat*. Il est possible d'extraire ou de modifier certaines pages, d'ajouter des commentaires ou de les crypter, etc.

Si la topographie a été produite en utilisant de données géoréférencées, elle contient également les informations de géoréférencement. Celles-ci peuvent être extraites par *XTherion* pour produire des images raster géoréférencées (voir *XTherion / Outils supplémentaires* pour plus de détails).

### **Topographies pour SIG (Systèmes d'Informations Géographiques)**

Les topographies peuvent être exportées aux formats DXF, ESRI ou KML afin d'être traitées ultérieurement dans les logiciels appropriés. Ces topographies ne contiennent pas le graphisme des symboles topo.

### **Topographies pour applications spécifiques**

Les topographies au format XVI contiennent la ligne de cheminement avec ses visées LRUD (et éventuellement des esquisses numérisées et morphées). Un fichier ayant ce format peut être importé dans *XTherion* pour servir d'arrière-plan à la numérisation de la topographie.

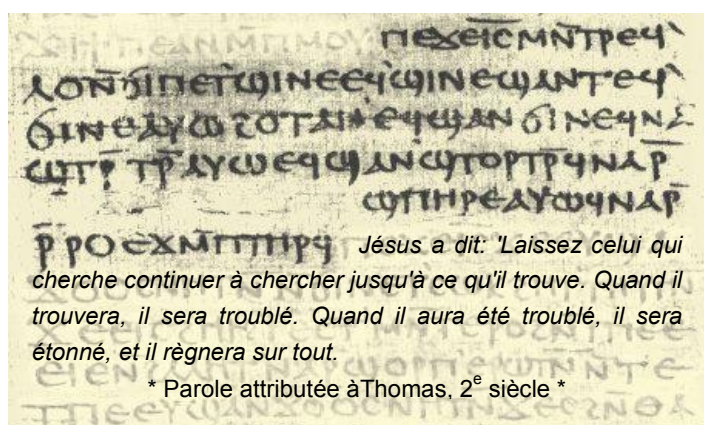
Les topographies au format *Survex* sont destinées à un aperçu rapide dans le logiciel 'Aven'.

### **Modèles 3D**

*Therion* peut exporter un modèle 3D dans différents formats en plus de son format natif *.lox*. Ceux-ci peuvent être chargés dans des programmes de visualisation, d'édition ou de raytracing afin de les imprimer ou de les traiter. Si le format ne prend pas en charge la définition des silhouettes du volume des galeries, seule la ligne de cheminement est incluse.

### **Loch**

Loch est un visualiseur de modèles 3D (fichiers *.lox*) inclus dans la distribution *Therion*. Il prend en charge par exemple le rendu en haute résolution du fichier et la visualisation stéréoscopique à l'aide de lunettes 3D.



### **Mise en page des dessins en PDF**

Ce chapitre est extrêmement utile si vous n'êtes pas satisfait de la mise en page prédéfinie (*layout*) des symboles et des topographies proposés et que vous souhaitez les adapter à vos besoins. Pour ce faire, vous devez toutefois être capables de comprendre et d'écrire les macros *T<sub>EX</sub>* et *METAPOST* standard.

### **Mise en page en mode Atlas**

La commande de présentation et de mise en page (*layout*) permet la configuration de base des pages en mode Atlas. Pour cela on utilise des options telles que *page-setup* ou *overlap*.

Mais il n'y a pas d'options qui spécifieraient la position de la topographie (*map*), de la fenêtre de navigation ni des autres éléments à l'intérieur de la zone définie par les dimensions `page-width` (largeur) et `page-height` (hauteur) de la page. Par exemple, pourquoi la fenêtre de navigation est-elle au-dessous de la topographie et non à sa droite ou à sa gauche ?

Il existe de nombreuses possibilités de mise en page de l'atlas. Aussi, plutôt que d'offrir encore plus d'options pour la commande `layout`, *Therion* utilise le langage T<sub>E</sub>X pour décrire d'autres types de mises en page. Cette approche présente l'avantage de permettre à l'utilisateur d'avoir un accès direct à un moteur de composition avancé sans pour autant surcharger le langage de *Therion*.

*Therion* utilise pdfT<sub>E</sub>X au format *plain* pour la composition d'une page. En conséquence, vous devrez vous familiariser avec le langage T<sub>E</sub>X si vous souhaitez définir de nouvelles mises en page (*layouts*).

La référence incontournable pour plain T<sub>E</sub>X est :

Knuth D. E, « The TEXbook, Reading, Massachusetts », Addison-Wesley 1984.

Pour les extensions de pdf pdfT<sub>E</sub>X T<sub>E</sub>X, il existe un petit manuel :

Thành, H. T., Rahtz S., Hagen H., "The pdfT<sub>E</sub>X user manual", disponible sur : <http://www.pdfTeX.org>

Les macros T<sub>E</sub>X sont utilisées dans la partie `code tex-atlas` de la commande de présentation `layout` (voir le chapitre « Traitement des données » pour plus de détails). La base prédéfinie par *Therion* est la macro `\dopage`.

L'idée est simple : *Therion* définit pour chaque page les variables T<sub>E</sub>X (`count`, `token` et `box registers`) qui contiennent les éléments de la page (`map`, `navigator`, nom de page, etc.). À la fin de chaque page, la macro `\dopage` est appelée. C'est elle qui définit la position de chaque élément sur la page. En redéfinissant cette macro, vous obtiendrez la mise en page souhaitée. Sans cette redéfinition, vous obtiendrez la mise en page standard.

Voici la liste des variables définies pour chaque page :

*Zones de mise en page (boxes) :*

- `\mapbox` ▷ espace contenant la topographie. Sa largeur (ou sa hauteur) sont définies en fonction des options de taille `size` et de chevauchement `overlap` de la commande de disposition `layout`.

`size_width + 2*overlap` ou

`size_height + 2*overlap`, respectivement

- `\navbox` ▷ zone contenant le navigateur, avec ses dimensions :  
`size_width * (2*nav_size_x+1) / nav_factor` ou  
`size_height * (2*nav_size_y+1) / nav_factor` respectivement

Les deux `\mapbox` et `\navbox` contiennent aussi des hyperliens.

*Nombre de registres :*

- `\pointerE`, `\pointerW`, `\pointerN`, `\pointerS` contient le numéro des pages voisines dans les directions E, W, N et S. Si aucune page de ce type n'existe, le numéro de page est défini sur 0.
- `\pagenum` numéro de page courante

*Apparence des registres :*

- `\pointerU`, `\pointerD` contient des informations sur les pages situées au-dessus et au-dessous de la page actuelle. Il consiste en un ou plusieurs enregistrements concaténés. Chaque enregistrement a un format spécial



`page-name|page-number|destination|`

Si de telles pages n'existent pas, la valeur est définie sur `notdef`.

Voir la description de la macro `\processpointeritem` ci-dessous pour savoir comment extraire et utiliser ces informations.

- `\pagename` ▷ nom de la topographie actuelle en fonction des options de la commande `map`.
- `\pagelabel` ▷ libellé de la page, spécifié par les options `origin` et `origin-label` de la commande `map`.

Les variables suivantes sont définies au début du document :

- `\hsize`, `\vsize` ▷ des pages T<sub>E</sub>X définies en fonction des paramètres `page-width` et `page-height` de l'option `page-setup` de la commande `layout`. Elles déterminent l'aire utile lorsque la mise en page est effectuée via la macro `\dopage`.
- `\ifpagenumbering` ▷ ce terme booléen (conditionnel) est vrai ou faux en fonction de l'option `page-numbers` de la commande `layout`.

Il existe également des macros prédéfinies qui facilitent le traitement des variables `\pointer*`:

- `\showpointer` avec l'un des arguments `\pointerE`, `\pointerW`, `\pointerN` ou `\pointerS` affiche la valeur de l'argument. Si cette valeur est 0, il n'affiche rien. C'est effectivement utile parce que la valeur zéro (pas de page voisine) ne doit pas être affichée.
- `\showpointerlist` avec l'un des arguments `\pointerU` ou `\pointerD` présente le contenu de cet argument. (qui contient `\pointerU` or `\pointerD`, voir ci-dessus.) Pour chaque enregistrement, il appelle la macro `\processpointeritem`, responsable du formatage des données.

La macro `\showpointerlist` doit être utilisée sans être redéfinie à l'endroit où vous souhaitez afficher le contenu de son argument. Pour obtenir un formatage personnalisé des données, redéfinissez la macro `\processpointeritem`.

- `\processpointeritem` possède trois arguments (`page-name`, `page-number`, `destination`) et permet de visualiser ces données. Les arguments sont délimités comme ci-dessous :

```
\def\processpointeritem#1|#2|#3\endarg{...}
```

Un exemple de définition pourrait être :

```
\def\processpointeritem#1|#2|#3\endarg{%
  \hbox{\pdfstartlink attr {/Border [0 0 0]}%
        goto name {#3} #2 (#1)\pdfendlink}%
}
```

(Notez comment utiliser l'argument de destination), ou de façon plus simple (si nous n'avons pas besoin des fonctionnalités des liens hypertextes) :

```
\def\processpointeritem#1|#2|#3\endarg{% \hbox{#2
  (#1)}%
}
```

Il existe aussi des macros pour la gestion des polices :

- `\size[#1]` pour les changements de taille.

- `\cmykcolor[#1 #2 #3 #4]`, `\[rgb]color[#1 #2 #3]` et `\graycolor[#1]` pour les changements de couleur du texte (les valeurs doivent être comprises entre 0 et 100),
- `\black` qui sélectionne la couleur noire dans le modèle de couleurs approprié,
- `\rm`, `\it`, `\bf`, `\ss`, `\si` pour modifier le formatage des caractères (normal, italique, gras,...).

Voir ci-dessous une liste de textes prédéfinis pouvant être utilisés dans l'atlas.

Il existe également une macro `\framed` qui prend une boîte comme argument et qui affiche le contour de cette boîte. Le style du contour peut être personnalisé en redéfinissant la macro `\linestyle` dont la valeur par défaut est `1 J 1 j 1.5 w`.

Nous sommes maintenant prêts à définir la macro `\dopage`. Vous pouvez choisir lequel des éléments prédéfinis vous souhaitez utiliser. Un exemple très simple pourrait être :

```
layout my_layout
  scale 1 200
  page-setup 29.7 21 27.7 19 1 1 cm
  size 26.7 18 cm
  overlap 0.5 cm
  code tex-atlas
    \def\dopage{\box\mapbox}
    \insertmaps
endlayout
```

Celui-ci définit la mise en page en mode paysage et au format A4, sans la fenêtre de navigation ni le texte. Il n'y a alors que la topographie sur la page.

Notez la macro `\insertmaps`. Les pages de topographies sont insérées à leur position. Ce n'est pas fait automatiquement parce que vous pouvez choisir d'insérer d'autres pages avant la première page de topographie.

La définition par défaut de la macro `\dopage` est plus complète :

```
\def\dopage{%
\ vbox{\centerline{\framed{\mapbox}}}
\bigskip
\line{%
\ vbox to \ht\ navbox{
\ hbox{\size[20]\the\pagelabel
\ifpagenumbering\space(\the\pagenum)\fi
\space\size[16]\the\pagename}
\ifpagenumbering
\medskip
\hbox{\qqquad\qqquad
\ vtop{%
\ hbox to Opt{\hss\showpointer\pointerN\hss}
\ hbox to Opt{\llap{\showpointer\pointerW\hskip0.7em}%
\raiselpt\hbox to Opt{\hss$\updownarrow$\hss}%
\raiselpt\hbox to Opt{\hss$\leftrightrightarrow$\hss}%
\rlap{\hskip0.7em\showpointer\pointerE}}
\hbox to Opt{\hss\showpointer\pointerS\hss}
}\qqquad\qqquad
\ vtop{
\def\arr{$\uparrow$}
\showpointerlist\pointerU
\def\arr{$\downarrow$}
\showpointerlist\pointerD
```

```

    }
  }
  \fi
  \vss
  \scalebar
}\hss
\box\navbox
}
}
}

```

En utilisant d'autres macros T<sub>E</sub>X ou des primitives T<sub>E</sub>X, il est possible d'ajouter d'autres fonctionnalités, par exemple : une mise en page différente pour les pages paires et impaires; des en-têtes et pieds de page ou bien ajouter un logo à chaque page.

Outre les pages de topographie, l'atlas contient des éléments supplémentaires : page de titre, informations de base sur la cavité, légende avec les symboles utilisés, etc.

*Therion* génère automatiquement une liste des symboles utilisés ainsi que la liste des personnes qui ont découvert, topographié et dessiné une partie sélectionnée de la cavité. Les registres (token) suivants peuvent être utilisés selon les besoins de l'utilisateur, avant ou après la macro `\insertmaps` :

- `\explotitle`, `\topotitle`, `\cartotitle` ▷ titres traduits
- `\exploteam`, `\topoteam`, `\cartoteam` ▷ membres participants (en rapport avec les options `team`, `explo-team` de la `centrelines` et avec l'option `author` des scraps)
- `\explodate`, `\topodate`, `\cartodate` ▷ dates correspondantes
- `\comment` ▷ défini par les options de `map-comment` de la mise en page (`layout`)
- `\copyrights` ▷ défini par les options de copyright des `surveys` et `scrap` et autres objets
- `\cavename` ▷ nom de la topographie exportée; défini en fonction des options `-title` de la topographie exportée
- `\cavelength`, `\cavedepth` ▷ longueur approximative et profondeur de la topographie affichée
- `\cavelengthtitle`, `\cavedepthtitle` ▷ étiquettes traduites
- `\cavemaxz`, `\caveminz` ▷ valeur max/min de l'altitude
- `\thversion` ▷ version actuelle de *Therion*
- `\currentdate` ▷ date courante
- `\outcscode`, `\outcsname` ▷ code et nom du système de coordonnées d'exportation
- `\northdir` ▷ 'true' (nord vrai) ou 'grid' (nord grille)
- `\magdecl` ▷ déclinaison magnétique en degrés

- `\gridconv` ▷ convergence des méridiens de la grille en degrés

Il existe aussi une macro `\atlastitlepages` qui combine la plupart des registres de symboles mentionnés ci-dessus afin d'obtenir des pages d'introduction simples pour un atlas préformaté.

Pour l'affichage de la légende il existe :

- `\iflegend` ▷ conditionnel ; vrai si l'option `legend` de la commande de mise en page `layout` a été définie sur les valeurs `on` ou `all`
- `\legendtitle` ▷ registre d'unités lexicales (token register) contenant le titre traduit de la légende
- `\insertlegend` ▷ macro pour l'insertion d'images de symboles de légende avec des descriptions traduites dans le nombre de colonnes spécifié (selon l'option de mise en page `legend-columns`)
- `\formattedlegend` ▷ combine les trois commandes ci-dessus pour obtenir une légende préformatée avec un en-tête et des symboles présentés sur deux <sup>70</sup>colonnes si l'option `legend` est sur `on`

La flèche indiquant le nord et la barre d'échelle peuvent être affichées en utilisant :

- `\ifnorthandarrow` ▷ conditionnel; vrai si la projection cartographique est en plan et que le symbole flèche-nord n'est pas caché dans la mise en page (`layout`)
- `\ifscalebar` ▷ conditionnel; vrai si la barre d'échelle n'est pas masquée
- `\northandarrow` ▷ formulaire PDF avec la flèche du nord
- `\scalebar` ▷ formulaire PDF avec une barre d'échelle

Il existe une macro à usage général pour la composition en plusieurs colonnes :

- `\begmulti <i>`, `\endmulti` ▷ le texte entre ces macros est composé en `<i>` colonnes.

Il existe une macro à usage général pour la composition en plusieurs colonnes <sup>71</sup>:

Exemple de création d'un atlas avec des listes de topographes, etc. suivies des pages de topos et d'une légende finale :

```
code tex-atlas
\atlastitlepages
\insertmaps
\formattedlegend
```

Pour utiliser un chemin relatif lors de l'inclusion d'un autre fichier T<sub>E</sub>X, utilisez la macro `\inputrel` au lieu de `\input`.

### Mise en page en mode topographie (cartographie)

<sup>70</sup> Deux par défaut ; ajustez l'option de mise en page (`layout`) des colonnes de la légende `legend-columns` pour en obtenir plus ou moins de deux.

<sup>71</sup> Cette macro ne doit pas être utilisée pour la légende d'une carte, sur celle-ci plusieurs colonnes doivent être ajustées par l'option de mise en page (`layout`) de `legend-columns`.

En mode topographie (cartographie), il est possible d'utiliser une grande partie des variables prédéfinies décrites dans le chapitre précédent :

```
\cavename, \comment, \copyrights, \exploitle, \topotitle, \cartotitle,  
\exploteam, \topoteam, \cartoteam, \exploodate, \topodate, \cartodate,  
\cavelength, \cavedepth, \cavelengthtitle, \cavedepthtitle, \cavemaxz,  
\caveminz, \thversion, \currentdate, \outcscod, \outcsname, \northdir,  
\magdecl, \gridconv, \ifnortharrow, \ifscalebar, \northarrow, \scalebar,  
\iflegend, \legendtitle, \insertlegend, \begmulti <i>, \endmulti,  
\formattedlegend, \legendcolumns.
```

Pour les placer quelque part sur la page de la carte, vous devez définir une macro `\maplayout` dans la section `code tex-map` de la commande de mise en page `layout`. Elle devrait contenir un ou plusieurs appels `\legendbox`. La macro `\legendbox` possède quatre paramètres: des coordonnées allant de 0 à 100, la spécification de l'alignement (N, E, S, W, NE, SE, SW, NW ou C) et le contenu à afficher.

En voici un exemple simple :

```
\def\maplayout{  
  \legendbox{0}{100}{NW}{\northarrow}  
}
```

Cet exemple affiche la flèche du nord dans le coin supérieur gauche de la feuille de topo.

Le quatrième paramètre peut également être une macro `\loadpicture`, qui inclut une image au format PDF, JPEG ou PNG. Bien qu'elle accepte un chemin d'accès relatif ou absolu, dans certains cas<sup>72</sup> il est nécessaire de convertir un chemin d'accès relatif en chemin d'accès absolu et de l'enregistrer pour une utilisation ultérieure :

```
\savepath{mypath1}{../data/picture.png}  
\def\maplayout{  
  \legendbox{100}{80}{NW}{\loadpicture{mypath1}}  
}
```

Pour la commodité de l'utilisateur, il existe un registre d'unités lexicales `\legendcontent`. Il contient le nom de la cavité préformaté, la flèche nord, la barre d'échelle, les équipes explo / topo / carto, les commentaires, les droits d'auteur et la légende. (`\legendcontent` est également utilisé dans la définition de présentation de carte par défaut :

```
\def\maplayout{\legendbox{0}{100}{NW}{\the\legendcontent}}).
```

La largeur du texte ci-dessus peut être ajustée par le registre de dimensions `\legendwidth` (sa valeur par défaut est définie par l'option de mise en page `legend-width`). La couleur et la taille des textes de la légende préformatée peuvent être facilement modifiées à l'aide des registres d'unités lexicales `\legendtextcolor`, `\legendtextsize`, `\legendtextsectionsiz` et `\legendtextheadersize`, par exemple pour obtenir un grand texte bleu :

```
code tex-map  
  \legendwidth=20cm  
  \legendtextcolor={\cmykcolor[0 30 50 50]}  
  \legendtextsize={\size[20]}  
  \legendtextheadersize={\size[60]}
```

Il est possible d'afficher l'ensemble de la topographie encadrée en réglant la dimension du registre `\framethickness` sur une valeur positive, par exemple. 0,5mm.

---

<sup>72</sup> Si vous combinez plusieurs mises en page stockées dans des répertoires différents.

## Personnalisation des étiquettes de texte

À partir de la version .1, vous pouvez utiliser l'option de présentation `fonts-setup` à la place de la macro METAPOST `fonts_setup ()`.

## Nouveaux symboles topographiques

La commande de mise en page de *Therion* permet de passer facilement d'un ensemble de symboles cartographiques prédéfinis à un autre. Si aucun symbole ou ensemble de symboles ne vous convient, il est possible de concevoir de nouveaux symboles cartographiques.

Cependant, cela nécessite la connaissance du langage METAPOST, qui est utilisé pour la visualisation des cartes. Il est décrit dans "Hobby, J. D.: A User's Manual for METAPOST", disponible à l'adresse :

<https://www.tug.org/docs/metapost/mpman.pdf>

L'utilisateur peut également bénéficier d'une référence complète au langage METAFONT, qui est assez similaire à METAPOST, voir Knuth, D. E.: "The METAFONTbook", Reading, Massachusetts, Addison-Wesley 1986

De nouveaux symboles peuvent être définis dans la section `code metapost` de la commande de mise en page `layout`. Cela permet d'ajouter facilement de nouveaux symboles au moment de l'exécution. Il est également possible d'ajouter des symboles de façon permanente en les compilant dans l'exécutable de *Therion* (voir l'annexe pour savoir comment faire).

Chaque symbole doit avoir un nom unique, qui se compose des éléments suivants :

- une des lettres 'p', 'l', 'a', 's' respectivement pour les symboles de point, ligne, zone ou spéciaux;
- un caractère de soulignement ;
- le type de symbole tel qu'il figure dans le chapitre Format des données, sans les tirets ;
- si le symbole a un sous-type, ajouter le caractère de soulignement et le sous-type ;
- le caractère de soulignement ;
- l'identifiant de l'ensemble de symboles en majuscules

Exemple : le nom standard d'un symbole de point 'écoulement d'eau' avec un sous-type 'permanent' dans l'ensemble 'MY' est `p_waterflow_permanent_MY`. Le nom standard des types de symboles définis par l'utilisateur ne doit pas inclure l'identifiant du jeu de symboles, par exemple `p_u_bat`.

Chaque nouveau symbole doit être enregistré par un appel de macro `initsymbol("<standard-name>")` ; à moins qu'il ne soit compilé dans l'exécutable *Therion*.

Il y a quatre plumes de traçage prédéfinies PenA (la plus épaisse) ... PenD (la plus fine), qui doivent être utilisés pour tous les dessins. Pour dessiner et remplir, utilisez les commandes `thdraw` et `thfill` au lieu des commandes `draw` and `fill` de METAPOST.

Les variables suivantes sont également disponibles <sup>73</sup>:

- boolean `ATTR__shotflag_splay`, `ATTR__shotflag_duplicate`, `ATTR__shotflag_approx` ▷ définies pour les lignes de la topographie
- boolean `ATTR__stationflag_splay` ▷ définie comme vraie pour les points situés aux extrémités des visées latérales d'habillage

---

<sup>73</sup> Si les noms entrent en conflit avec des commandes de *Therion* (comme `color`), vous pouvez ajouter un point d'exclamation '!' pour empêcher *Therion* d'analyser la ligne : `! color myNewColorDef`

- boolean `ATTR__scrap_centerline` ▷ définie comme vraie pour les calques *Therion* (scraps) créés à partir de la ligne centrale de cheminement
- boolean `ATTR__elevation` ▷ définie comme est vraie pour la coupe (étendue), fausse pour le plan
- numeric `ATTR__height` ▷ profondeur d'un puits ou hauteur de la paroi d'un puits
- string `ATTR__id` ▷ contient l'ID de l'objet courant
- string `ATTR__survey` ▷ contient le nom de la topographie en cours
- string `ATTR__scrap` ▷ contient le nom du claque *Therion* (scrap) en cours
- picture `ATTR__text` ▷ contient du texte saisi au clavier, par exemple pour la suite d'un point
- string `NorthDir` ▷ nord 'vrai' ou nord de la 'grille'
- numeric `MagDecl` ▷ déclinaison magnétique en degrés
- numeric `GridConv` ▷ convergence des méridiens de la grille en degrés

Si vous devez inclure certaines définitions de METAPOST à partir d'un fichier spécifié par un chemin relatif, vous devez utiliser la macro `inputrel("relative/path.mp")`

### **Symboles de points**

Les symboles de points sont définis comme des macros à l'aide des commandes `def ... enddef`. La plupart des définitions de symboles de points ont quatre arguments : position (couple), rotation (numérique), échelle (numérique) et alignement (couple). Les exceptions sont *section* qui n'a pas de représentation visuelle, *labels*, qui nécessite un traitement spécial comme décrit dans le chapitre précédent, et *station* qui ne prend qu'un seul argument : position (couple).

Tous les symboles de points sont dessinés en coordonnées locales avec l'unité de longueur *u*. Les plages recommandées sont -0,5*u*, 0,5*u* dans les deux axes. Le symbole doit être centré à l'origine des coordonnées. Pour la carte *Therion* finale, tous les dessins sont transformés comme spécifié dans la variable de transformation *T*, il est donc nécessaire de définir cette variable avant de dessiner.

Cela se fait généralement en deux étapes (supposons que les quatre arguments soient *P*, *R*, *S*, *A*) :

1. fixer la variable de la paire *U* à  $\left(\frac{width}{2}, \frac{height}{2}\right)$  du symbole pour un alignement correct.

L'argument d'alignement *A* est une paire représentant les rapports  $\left(\frac{shift_x}{U_x}\right)$  et  $\left(\frac{shift_y}{U_y}\right)$  (Par conséquent, `aligned A` signifie `shifted (xpart A * xpart U, ypart A * ypart U)`.)

2. définir la variable de transformation *T*: `=identity aligned A rotated R scaled S shifted P;`

Pour dessiner et remplir, utilisez les commandes `thdraw` et `thfill` au lieu des commandes `draw` et `fill` de METAPOST. Elles prennent automatiquement en charge la transformation *T*.

Voici un exemple de définition :

```
def p_entrance_UIS (expr P,R,S,A)=  
    U:=(.2u,.5u);  
  
    T :=identity aligned A rotated R scaled S shifted P;  
    thfill (-.2u,-.5u)--(0,.5u)--(.2u,-.5u)--cycle;  
  
enddef;  
  
initsymbol("p_entrance_UIS");
```

### **Symboles de lignes**

Les symboles de ligne diffèrent des symboles de point car ils ne possèdent pas de système de coordonnées local. Chaque symbole de ligne obtient le chemin en coordonnées absolues comme premier argument. Par conséquent, il est nécessaire de définir la variable T sur `identity` avant de dessiner.

Les symboles suivants prennent des valeurs supplémentaires *Arguments* :

- **arrow** > numérique : 0 pas de flèche, 1 flèche à la fin, 2 flèche au début, 3 flèche aux deux extrémités
- **contour** > texte : liste des points à cocher (tick) ou l'un des nombres suivants -1, -2 ou -3 respectivement pour indiquer une coche indéfinie, une coche au milieu ou aucune coche.
- **section** > texte : liste des points ayant une flèche d'orientation ou -1 pour indiquer l'absence de flèche
- **slope** > numérique : 0 sans bordure, 1 bordure ; texte : une liste de triplets (point,direction, longueur)

Exemple d'utilisation :

```
def l_wall_bedrock_UIS (expr P) =  
    T :=identity;  
  
    pickup PenA;  
  
    thdraw P;  
  
enddef;  
  
initsymbol("l_wall_bedrock_UIS");
```

### **Symboles de surfaces**

Les zones sont similaires aux lignes : elles ne prennent qu'un seul argument - `path` en coordonnées absolues.

Vous pouvez les compléter de trois manières :

- remplir une grille uniforme ou aléatoire dans une image temporaire (ayant des dimensions `bbox path`) avec des symboles ponctuels; coupez-la en fonction du `path` et ajoutez-la à `currentpicture`



- remplir `path` avec une couleur unie
- remplir `path` avec un motif prédéfini à l'aide d'un mot clé `withpattern`.
- remplir une grille uniforme ou aléatoire dans une image temporaire (ayant des dimensions `bbox path`) avec des symboles ponctuels; coupez-la en fonction du `path` et ajoutez-la à `currentpicture`

Les motifs sont définis en utilisant la même interface utilisateur (sans la macro `patterncolor`) que celle décrite dans l'article de Bolek, P.: "METAPOST and patterns," TUGboat, 3, XIX (1998), pp. 276–283, disponible en ligne à l'adresse <https://www.tug.org/TUGboat/Articles/tb19-3/tb60bolek.pdf>

Vous pouvez utiliser le terme standard METAPOST `draw` et des macros similaires sans définir la variable `T` dans les définitions des motifs. Dans les documents PDF, tous les motifs ne sont pas colorés, ce qui signifie que toute information de couleur spécifiée dans la définition du motif est ignorée et que la couleur est attribuée ultérieurement lorsque le motif est utilisé (à l'aide de l'option de disposition `symbol-color`).

Voici un exemple sur la façon de définir et d'utiliser des modèles :

```
beginpattern(pattern_water_UIS);
  draw origin--10up withpen pensquare scaled (0.02u);
  patternxstep(.18u);
  patterntransform(identity rotated 45);
endpattern;

def a_water_UIS (expr p) =
  T :=identity;
  thclean p;
  thfill p withpattern pattern_water_UIS; enddef;
initsymbol("a_water_UIS");
```

### **Symboles spéciaux**

Il existe actuellement deux symboles spéciaux : la barre d'échelle et la flèche indiquant le nord. Les deux sont expérimentaux et sujets au changement.

*1. Lorsqu'un scientifique distingué mais âgé affirme que quelque chose est possible, il a très certainement raison. Lorsqu'il déclare que quelque chose est impossible, il a très probablement tort.*

*2. La seule façon de découvrir les limites du possible est de s'aventurer un peu plus loin dans l'impossible.*

*3. Toute technologie suffisamment avancée est indiscernable de la magie.*

\* Arthur C. Clarke, 1973 \*

## **Annexe**

### **Compilation**

#### **Installation des dépendances**

Si vous voulez compiler *Therion* à partir du code source et l'exécuter, vous avez besoin des logiciels suivants (les trois premiers ne sont requis que pendant la compilation) :

- GNU C/C++ compiler
- GNU make

- *Perl*
- *Python 3*
- *PROJ library* (<https://proj.org/>) Les versions supportées sont : v4 : 4.9.3 ; v5 : 5.1.0 et plus récentes ; v6 : 6.2.1 et plus récente ; v7 : 7.0.0 et plus récente.
- *Tcl/Tk 8.4.3* et plus récents (<https://www.tcl.tk>) avec un widget *BWidget* défini (<https://sourceforge.net/projects/tcllib/>) et éventuellement l'extension *tkImg* (<https://sourceforge.net/projects/tkimg/>).
- une distribution  $T_E X$  comprenant a minima  $T_E X$  avec Plain format, le récent  $pdfT_E X$ , et *METAPOST* (<https://www.tug.org>).
- *LCDF Typetools package* (<https://www.lcdf.org/type/>)
- Une distribution *ImageMagick* avec des utilitaires de conversion et d'identification, si vous voulez utiliser la déformation des croquis de la topographie.
- *Ghostscript* si vous voulez créer des images calibrées à partir de cartes PDF géoréférencées.

Pour compiler *Loch*, il vous faut :

- *freetype 2* et plus récents; *freetype-config* doit fonctionner
- *wxWidgets 2.6* et plus récents; *wx-config* doit fonctionner
- *VTK 5.0* et plus récents
- *libjpeg*, *libpng*, *zlib*

Tous ces programmes (à l'exception de *BWidget* et du paquet *tkImg*) sont généralement inclus dans les distributions *Linux*, *Unix* ou *MacOS X*. Pour *Windows*, envisagez d'utiliser *MinGW* et *MSYS* (<http://www.mingw.org>). Il s'agit d'une distribution d'utilitaires GNU avec GNU make et GCC. (d'ailleurs pourquoi ne pas utiliser la version précompilée de *Windows* ?).

Voici un exemple d'installation des dépendances pour compiler *Therion* sur Ubuntu 18.04 ou 20.04 : `sudo apt install bwidget cmake gcc ghostscript imagemagick lcdftypetools libfreetype6-dev libjpeg-dev libpng-dev libproj-dev libtk-img-dev libvtk7-dev libwxgtk3.0-gtk3-dev tcl-dev texlive-binaries texlive-metapost zlib1g-dev`.

## Utilisation de CMake

Décompressez la distribution source `therion-5.*.tar.gz` et créez un répertoire séparé pour la construction, par exemple :

- `cd therion && mkdir build && cd build`
- `cmake [paramètres] ...`
- `make -j4`

## Paramètres CMake

Voici une sélection de paramètres qui peuvent être utilisés avec `cmake` :

- `-G <generator>` ▸ spécifie le générateur, par exemple `-G Ninja` pour utiliser le système de construction Ninja au lieu de make, ou `-G "MSYS Makefiles"` pour construire en utilisant make sous MSYS2
- `-DUSE_BUNDLED_SHAPELIB=OFF` ▸ utilise la bibliothèque Shapelib du système
- `-DUSE_BUNDLED_CATCH2=OFF` ▸ utilise la bibliothèque Catch2 du système
- `-DUSE_BUNDLED_FMT=OFF` ▸ utilise la bibliothèque fmt du système
- `-DECM_ENABLE_SANITIZERS=<option>` ▸ utilise les désinfectants d'exécution, options pertinentes : `address` qui détecte les accès mémoire invalides, les erreurs de type « use-after-free », les erreurs de type « double free », les pertes mémoire (utile pour le débogage) `undefined` qui détecte les comportements non définis, par exemple l'utilisation de valeurs non initialisées.
- `-DCMAKE_BUILD_TYPE=<option>` ▸ type de compilation, options pertinentes :
  - `Debug` - compile avec des symboles de débogage et des assertions
  - `Release` - compile avec des optimisations
  - `RelWithDebInfo` - compile avec des symboles de débogage et des optimisations, utiles pour le profilage et le débogage
- `-DTHBOOK_FORMAT=<option>` ▸ définit la taille de sortie du livre *Therion* (la plupart des images sont omises pour les petites tailles) ; options pertinentes :
  - 0 - format portrait A4
  - 1 - format portrait petit écran (certaines lignes et images ne rentreront pas)
  - 2 - format paysage petit écran
  - 3 - format optimisé pour les lecteurs d'ebook

### **Approche traditionnelle : utilisation de make**

- décompresser la distribution source `Therion-5.*.tar.gz`
- `cd Therion`
- `make config-macosx` or `make config-win32`, si vous utilisez respectivement MacOS X ou Windows
- `make`
- `sudo make install`

### **Paramètres du makefile**

Le makefile de *Therion* peut prendre des paramètres optionnels.

- `config-linux`, `config-macosx`, `config-win32` ▸ configure *Therion* pour une plateforme spécifique. Linux est une valeur par défaut.
- `config-release`, `config-oxygen`, `config-ozone` ▸ définir le niveau d'optimisation du compilateur C++ (none, `-O2` et `-O3`)
- `config-debug` ▸ utile avant de déboguer le programme
- `install` ▸ installe *Therion*
- `clean` ▸ supprime tous les fichiers temporaires

## Le guide du Hacker

### Cross-compilation pour Windows

*Therion* prend en charge la compilation d'exécutables *Win32* sous *Linux* à l'aide du compilateur croisé *MXE*. (<http://mxe.cc>).

- installer les paquets *static/win32* suivants (*i686-w64-mingw32.static-\**) ou les paquets *static/win64* (*x86-64-w64-mingw32.static-\**) dans le répertoire `/usr/lib/mxe/` : *binutils*, *bzip2*, *expat*, *freetype-bootstrap*, *gcc*, *gettext*, *glib*, *harfbuzz*, *jpeg*, *libiconv*, *libpng*, *proj*, *tiff*, *vtk*, *wxwidgets*, *xz*, *zlib*.
- modifier le `PATH`: `export PATH=/usr/lib/mxe/usr/bin:$PATH`
- utiliser *CMake* ou l'approche traditionnelle (`cd therion && make config-win32cross && make`) pour construire *Therion*.

Voir [therion/github/workflows/](https://github.com/therion/workflows/) pour des exemples détaillés de construction de *Therion* sur plusieurs plateformes.

### Ajout de nouvelles traductions

*Therion* prend en charge la traduction des étiquettes de titre des cartes. Supposons que vous souhaitiez ajouter une nouvelle langue `xx` :

- lancer `'perl process.pl export xx'` dans le sous-répertoire des sources de *Therion* 'thlang'. Cela crée un fichier `texts_xx.txt`. Ce fichier est codé en UTF-8.
- Éditer les textes `texts_xx.txt` file. Ajoutez vos traductions aux lignes commençant par `'xx:'`.
- lancez `make update`
- compilez *Therion*

### Ajout de nouveaux encodages

Bien que le codage Unicode UTF-8 couvre tous les caractères que *Therion* est capable de traiter, il peut être gênant de l'utiliser. Dans ce cas, il est possible d'ajouter une prise en charge de tout codage sur 8 bits pour les fichiers d'entrée de texte. Copier un fichier de traduction dans le répertoire `thchencdata`; ajoutez son nom au champ `"ifiles"` au début du script Perl `generate.pl`; lancez-le et recompilez *Therion*.

Le fichier de traduction doit contenir deux valeurs hexadécimales d'un caractère (la première avec le codage à 8 bits, la deuxième avec le codage Unicode) dans chaque ligne. Les commentaires possibles suivent le caractère `"#"`.

### Ajout de nouveaux encodages TEX

Il est facile d'ajouter de nouveaux encodages pour la sortie de carte 2D<sup>74</sup>. Copiez un fichier de mappage d'encodage approprié avec une extension `*.enc` dans `texenc/encodings`, exécutez le script Perl `mktextenc.pl` situé dans le répertoire `texenc` et compilez *Therion*.

*Therion* utilise les mêmes fichiers d'encodage que le programme `afm2tfm` de la distribution TEX, qui a le même format qu'un vecteur d'encodage dans une police PostScript. Vous trouverez plus de détails au chapitre 6.3.1.5 « Format du fichier de codage » dans la documentation du programme Dvips.

---

<sup>74</sup> Cette section s'applique à l'ancien style de sélection des polices à l'aide de la commande `tex-fonts` dans le fichier d'initialisation et est obsolète lorsque l'on utilise la commande `pdf-fonts`.

## Génération de nouveaux en-têtes TEX et METAPOST

*Therion* utilise T<sub>E</sub>X et METAPOST pour la visualisation et la composition de topos en 2D. Les macros prédéfinies sont compilées dans l'exécutable *Therion* et sont copiées dans le répertoire de travail juste avant l'exécution de METAPOST et T<sub>E</sub>X (sauf si l'option `--use-extern-libs` est utilisée). La commande `Mise en forme` permet de modifier certaines macros dans le fichier de configuration au moment de l'exécution.

Cependant, il est possible d'apporter des modifications permanentes aux fichiers de macros. Après avoir modifié les fichiers des répertoires `mpost` et `tex`, il est nécessaire d'exécuter les scripts Perl `genmpost.pl` et `gentex.pl`, qui génèrent des fichiers d'en-tête C ++, puis de compiler à nouveau l'exécutable *Therion*.

### Mise à jour du modèle géomagnétique

*Therion* utilise le modèle IGRF pour calculer la déclinaison magnétique. Téléchargez le modèle au format txt sur <https://www.ngdc.noaa.gov/IAGA/vmod/igrf.html> et sauvegardez-le dans le répertoire `geomag/` (par exemple `igrfXYcoeffs.txt`). Exécutez `./igrf2c.py igrfXYcoeffs.txt` ce qui créera `thgeomagdata.h` dans le répertoire source de *Therion* et recompilez *Therion*.

Pour tester le modèle, extrayez le fichier `sample_out_IGRFXY.txt`, qui est inclus dans la distribution *Geomag* disponible sur la même page web. Placez-la dans le répertoire `geomag/test/`, exécutez `./build.sh sample_out_IGRFXY.txt` et vérifiez les lignes ayant un point d'exclamation dans la sortie.

## Variables d'environnement

*Therion* lit les variables d'environnement suivantes :

- `THERION` ▷ [non requis] recherche le chemin pour le(s) fichier(s) (x)*Therion*.ini
- `HOME` (`HOMEDRIVE` + `HOMEPath` sur WinXP) ▷ [non requis, mais généralement présent sur votre système] recherche le chemin pour le(s) fichier(s) (x)*Therion*.ini
- `TEMP`, `TMP` ▷ répertoire temporaire du système, où *Therion* stocke les fichiers temporaires (dans un répertoire nommé `th$PID$,` où `$PID$` est un process ID), sauf si `tmp-path` est spécifié dans le fichier d'initialisation.

Consultez la documentation de votre système d'exploitation pour savoir comment les configurer.

## Fichiers d'initialisation

Les paramètres dépendant du système de *Therion* et de *XTherion* sont spécifiés respectivement dans les fichiers *Therion*.ini ou *XTherion*.ini. Ils sont placés dans les répertoires suivants :

- pour UNIX: `., $THERION, $HOME/.Therion, /etc, /usr/etc, /usr/local/etc`
- pour Windows : `., $THERION, $HOME\Therion, <Therion-installation-directory>, C:\WINDOWS, C:\WINNT, C:\Program Files\Therion`

### *Therion*

Si aucun fichier n'est trouvé, *Therion* utilise ses paramètres par défaut. Si vous voulez les lister, utilisez l'option `--print-init-file`. Le fichier d'initialisation est lu comme n'importe quel autre fichier *Therion*. (Les lignes vides ou commençant par '#' sont ignorées; les lignes se terminant par une barre oblique

inversée se poursuivent à la ligne suivante.) Voici les commandes d'initialisation actuellement prises en charge :

- `loop-closure <Therion/Survex>`

Par défaut, *Survex* est utilisé s'il est présent, sinon ce sera *Therion*.

- `encoding-default <encoding-name>`

Définissez le codage de sortie par défaut (actuellement non utilisé).

- `encoding-sql <encoding-name>`

Définissez le codage de sortie par défaut pour l'exportation SQL.

- `language <xx[_YY]>`

Langue de sortie par défaut. Voir la page de copyright pour la liste des langues disponibles.

- `units <metric/imperial>` Définir les unités par défaut.

- `mpost-path <file-path>`

Définir le chemin complet d'un exécutable METAPOST si *Therion* ne le trouve pas. ("`mpost`" par défaut).

- `mpost-options <blabla>` Définir les options METAPOST.

- `pdftex-path <file-path>`

Définir le chemin complet d'un exécutable pdf<sub>T</sub>E<sub>X</sub> si *Therion* ne le trouve pas ("`pdfetex`" par défaut).

- `identify-path <file-path>`

Définir le chemin complet d'un exécutable ImageMagick si *Therion* ne le trouve pas ("`identify`" par défaut).

- `convert-path <file-path>`

Définir le chemin complet de l'exécutable de conversion ImageMagick si *Therion* ne parvient pas à le trouver. ("`convert`" par défaut).

- `source-path <directory>`

Chemin vers les fichiers de données et de configuration. Utilisé principalement pour les niveaux d'échelle du système et les définitions de mise en page (`layout`).

- `tmp-path <directory>`

Chemin où le répertoire temporaire doit être créé.

- `tmp-remove <OS command>`

Commande système pour supprimer des fichiers du répertoire temporaire.

- `tex-env <on/off>`

[Fonctionne sous Windows uniquement.] Lorsque cette option est désactivée (par défaut), *Therion* efface temporairement toutes les variables d'environnement liées à TEX. C'est utile si une autre

distribution T<sub>E</sub>X est installée sur votre système et qu'elle a déjà configuré toutes les variables d'environnement, ce qui risquerait de perturber les programmes T<sub>E</sub>X et METAPOST fournis dans la distribution *Therion* pour Windows.

Activez cette option si vous utilisez une autre distribution T<sub>E</sub>X pour le traitement des topographies.

- `text <language ID> <Therion text> <my text>`

En utilisant cette option, vous pouvez modifier n'importe quelle traduction par défaut de texte *Therion* dans un fichier de sortie. Pour une liste des textes *Therion* et des traductions disponibles, voir le fichier `thlang/texts.txt`.

- `cs-def <id> <proj4def>`

Définissez un nouveau système de coordonnées `<id>` en utilisant la syntaxe Proj4.

- `proj-auto <on/off>`

Si cette option est activée, PROJ v6+ décide de la transformation optimale<sup>75</sup> entre les systèmes de coordonnées. Les transformations sélectionnées sont listées dans le fichier log. Il est recommandé de spécifier les systèmes de coordonnées en utilisant directement les codes EPSG (par exemple EPSG:4258). Si cette option est désactivée (`off`) ou si *Therion* utilise un PROJ antérieur à la version 6, les coordonnées de la source sont d'abord transformées en `wgs-84`, puis de `wgs-84` vers le système de coordonnées cible. Ceci peut entraîner une précision médiocre. Le paramètre par défaut est désactivé (`off`).

- `proj-missing-grid <ignore/warn/fail/cache/download>`

Définit la gestion des grilles de transformation manquantes si `proj-auto` est activé. Les grilles sont utilisées pour obtenir une meilleure précision de transformation entre certains systèmes de coordonnées.

`ignore` permet à Proj de choisir une autre transformation qui n'utilise pas les grilles manquantes ; cela conduit généralement à une diminution de la précision de la transformation (par exemple en mètres au lieu de centimètres). Voir le fichier log pour une liste des transformations utilisées et leurs précisions.

`warn` se comporte comme `ignore`, mais imprime des avertissements sur les grilles manquantes (les liens de téléchargement sont généralement affichés aussi<sup>76</sup>).

`fail` s'arrête après la détection de la première grille manquante et affiche le lien de téléchargement.

`cache` active la connectivité du réseau et permet à Proj de télécharger les parties manquantes des grilles depuis Internet. Les informations sont stockées dans un cache local<sup>77</sup>. Comme seules les parties de grilles couvrant la zone actuelle sont téléchargées, c'est potentiellement plus rapide et moins consommateur d'espace que le téléchargement. L'inconvénient est que le cache local n'est pas utilisé par Proj si le réseau est désactivé (dans *Therion*, vous devez utiliser le mode cache pour utiliser le local).

`download` permet d'activer temporairement la connectivité réseau et permet à Proj de télécharger les grilles manquantes depuis l'Internet. Les grilles sont utilisées dans les exécutions suivantes (dans n'importe quel mode).

Le paramètre par défaut est `warn`. L'option `proj-missing-grid` est ignorée si `proj-auto` est désactivé ou si Proj v5 ou plus ancien est utilisé. Si Proj v6 est utilisé par *Therion*, les paramètres `cache` et `download` sont en échec, car la connectivité réseau est une fonctionnalité de Proj7 et plus.

---

<sup>75</sup> Dans ce cas, la fonction `proj create crs to crs()` est utilisée. Sinon *Therion* appelle la fonction `proj create()` de la bibliothèque PROJ.

<sup>76</sup> Voir [https://proj.org/resource\\_files.html](https://proj.org/resource_files.html) pour les instructions où placer les grilles téléchargées.

<sup>77</sup> Voir <https://proj.org/usage/network.html>

- `pdf-fonts <rm> <it> <bf> <ss> <si>`

Configurez les polices à utiliser sur les fichiers PDF. La commande doit être suivie de chemins spécifiant l'emplacement des polices régulières, italiques, gras, sans-serif et sans-serif oblique dans votre système. Les polices TrueType et OpenType sont prises en charge.

*Therion* exige que LCDF Typetools soit installé sur votre système pour pouvoir utiliser cette commande.

Exemple :

```
pdf-fonts  "/usr/share/fonts/Serif.ttf" \
"/usr/share/fonts/Serif-Italic.ttf" \
"/usr/share/fonts/Serif-Bold.ttf" \
"/usr/share/fonts/Sans.ttf" \
"/usr/share/fonts/Sans-Oblique.ttf"
```

- `otf2pfb <on/off>`

Lorsque cette option est activée (`on` valeur par défaut), les polices OpenType utilisées dans `pdf-fonts` sont converties en polices PFB, si elles sont basées sur PostScript. Certaines informations sont perdues au format PFB, mais il est intéressant que pdf<sub>T</sub>E<sub>X</sub> puisse incorporer un sous-ensemble de polices PFB (par opposition aux polices OpenType qui doivent être entièrement intégrées).

- `tex-fonts <encoding> <rm> <it> <bf> <ss> <si>`

Méthode originale et plus compliquée pour configurer les polices pour les cartes PDF. Vous devez spécifier explicitement l'encodage (maximum 256 caractères de la police qui sera utilisée). La liste des encodages actuellement supportés donne l'option de ligne de commande `--print-tex-encodings`. Le même encodage doit être utilisé lors de la génération de métriques T<sub>E</sub>X (fichiers \*.tfm) pour ces polices (par exemple avec le programme `afm2tfm`) et cet encodage doit être explicitement donné aussi dans le fichier map de pdf<sub>T</sub>E<sub>X</sub>. La seule exception est le jeu de base des polices Computer Modern, qui utilise un encodage "brut". Cet encodage n'a pas besoin d'être spécifié dans le fichier map de pdf<sub>T</sub>E<sub>X</sub>.

L'encodage doit être suivi de cinq spécifications de police pour les styles normal, italique, gras, Sans Serif et Sans Serif oblique. Le paramètre par défaut est `tex-fonts raw cmr10 cmti10 cmbx10 cmss10 cmssi10`

Exemple comment utiliser d'autres polices (par exemple TrueType Palatino dans l'encodage `xl2` (un encodage dérivé de ISO8859-2)). Exécuter :

```
ttf2afm -e xl2.enc -o palatino.afm palatino.ttf
afm2tfm palatino.afm -u -v vpalatino -T xl2.enc
vptovf vpalatino.vpl vpalatino.vf vpalatino.tfm
```

Vous obtenez des fichiers `vpalatino.vf`, `vpalatino.tfm` et `palatino.tfm`.

Ajoutez la ligne `palatino <xl2.enc <palatino.ttf` au fichier de carte du pdf<sub>T</sub>E<sub>X</sub>. Il faut faire de même pour les caractères italiques et gras et les polices Sans Serif et Sans Serif oblique correspondantes.

Si vous êtes paresseux, essayez ceci :

```
tex-fonts xl2 palatino palatino palatino palatino palatino
```



(Nous devrions utiliser la police virtuelle actuelle `vpalatino` au lieu de `palatino`, qui ne contient ni crénage ni ligatures, mais pdfT<sub>E</sub>X ne supporte pas la commande `\pdfincludechars` sur les polices virtuelles. A améliorer).

Si vous voulez ajouter des encodages non supportés, lisez le chapitre Compilation / Guide du hacker.

- `tex-fonts-optional <encoding> <rm> <it> <bf> <ss> <si>`

Similaire à `tex-fonts`, mais teste si les polices T<sub>E</sub>X sont installées sur le système. Il ne fait rien si l'une des polices spécifiées n'est pas présente.

Ce paramètre est utilisé par défaut pour les polices tchèque/slovaque et cyrillique pour éviter les erreurs METAPOST sur les systèmes sans ces polices présentes.

Comme le test prend un certain temps (l'instance de pdfT<sub>E</sub>X est lancée), vous pouvez désactiver complètement le comportement par défaut en définissant `tex-fonts` dans le fichier INI.

- `tex-refs-registers <on/off>`

Basculez entre l'utilisation des registres de comptage et des macros pour stocker les références aux objets graphiques dans T<sub>E</sub>X. Chacun ayant ses propres avantages, voir la section Limitations.

## ***XTherion***

Le fichier d'initialisation pour *XTherion* est en réalité un script Tcl évalué au démarrage de *XTherion*. Le fichier est commenté; voir les commentaires pour plus de détails.

### **Limitations**

- taille des segments de calque (`scraps`) ▷  
METAPOST en mode par défaut ('`scaled`') ≈ 2.8 × 2.8 m à l'échelle de sortie  
METAPOST en mode "`double`"<sup>78</sup> : pratiquement aucune limite<sup>79</sup>
- taille de la page ▷  
topo PDF pour atlas : ≈ 5 × 5 m (pdfT<sub>E</sub>X limit)  
topo SVG: sans limite
- nombre de segments de calque (`scraps`) ▷ approx. 500–6000, dépend du nombre des sections transverses  
limite METAPOST standard : 4 (scraps + sections) < 4096 (peut être augmenté arbitrairement)  
limite pdfT<sub>E</sub>X : 2 × pages + images + modèles + 6 (scraps + sections) < 32500

---

<sup>78</sup> Pour lancer METAPOST dans le mode '`double`', `mpost-options "-numbersystem=double"` dans le fichier d'initialisation. Il n'est pas recommandé d'utiliser les modes de précision arbitraire '`decimal`' et '`binary`', car il y a encore des bogues dans leur implémentation et ils sont beaucoup plus lents que le mode '`double`'. Vous devez avoir un METAPOST postérieur à la version 2.00 pour utiliser ce mode sans problème.

<sup>79</sup> Ce mode est assez puissant pour que ce soit possible.

## Exemple de données

L'exemple suivant illustre l'utilisation de base des commandes *Therion* :

```
encoding utf-8

survey main -title "Test cave"

    survey first
        centreline
            units compass grad
            data normal from to compass clino length
                1 2      100 -5  10
    endcentreline
endsurvey

survey second -declination [3 deg]
    centreline
        calibrate length 0 0.96
        data normal from to compass length clino
            1 2 0      10  +10
    endcentreline
endsurvey

centreline
    equate 2@first 1@second
endcentreline

# scraps are usually in separate *.th2 files scrap
s1 -author 2004 "Therion team"

    point 763 746 station -name 2@second
    point 702 430 station -name 2@first
    point 352 469 station -name 1@first
    point 675 585 air-draught -orientation 240 -scale large

    line wall -close on
        287 475
        281 354 687 331 755 367
        981 486 846 879 683 739
        476 561 293 611 287 475
    endline

endscrap

map m1 -title "Test map"
    s1
endmap

endsurvey
```

Le fichier de configuration correspondant pourrait être :

```
encoding utf-8

source test

layout l1
```

```
scale 1 100

layers off

endlayout

select m1@main

export model -fmt Survex

export map -layout l1
```

Si vous enregistrez le fichier de données sous le nom «[test.th](#)» et le fichier de configuration sous le nom «*thconfig*», vous pouvez les traiter avec *Therion* pour faire un essai.

## Historique

### • 1999

Octobre : premières idées concrètes.

Novembre : début de la programmation (scripts Perl et macros METAPOST).

27 décembre : *Therion* compile pour la première fois une topo simple au format PostScript (32 Ko de Perl et 7 Ko de code source METAPOST et TEX). Le modèle de déformation de la topographie était substantiellement différent du modèle actuel (la position des entités était relative à une visée particulière et non à la position de toutes les stations dans un segment). Cette version incluait déjà certaines fonctionnalités intéressantes, telles que les fonctions de transformation qui permettaient à l'utilisateur de spécifier le format de saisie des données du relevé topo ou de scinder de grandes topos en plusieurs feuilles.

30 décembre : première page Web (avec des exemples de données mais sans code source).

### • 2000

Janvier : xthedit (Tcl / Tk), une interface graphique pour *Therion*

18 février : début de la reprogrammation (Perl)

1er avril : le premier hyperlien PDF topographies de grottes / atlas

Août : expériences avec PDF, pdfT<sub>E</sub>X et METAPOST

### • 2001

Novembre : début de la réimplémentation à partir de zéro : *Therion* (C ++ avec certains scripts Perl hérités de la version précédente); notion de segment de calque (*scrap*); éditeur de carte 2D interactif ThEdit en remplacement de xthedit (Delphi).

Décembre : ThEdit exporte une topographie simple pour la première fois.

### • 2002

Mars : *Therion 0.1* - *Therion* est capable de traiter les données d'un relevé (ligne de cheminement) de la « Grotte des chauves-souris mortes ». *XTherion*, éditeur de texte conçu pour *Therion* (Tcl / Tk).

27 juillet : *Therion 0.2* - *Therion* compile pour la première fois une topo simple composée de deux segments (*scraps*) (800 Ko de code source).

Août : *XTherion* étendu à l'éditeur de carte 2D (en remplacement de ThEdit)

Septembre : *Therion* compile la première topo réelle et complexe d'une grotte. *XTherion* étendu au compilateur.

- **2003**

Mars : la première version du livre *Therion* est terminée.

Avril : *Therion* inclus dans Debian GNU / Linux.

Juin : tous les scripts Perl ont été réécrits en C ++, *Therion* est maintenant un programme exécutable (bien qu'utilisant *Survex* et TEX).

- **2004**

Mars : *Therion* 0.3 - *Therion* exporte un modèle 3D créé à partir de topographies en 2D. Algorithme de fermeture de boucle inclus dans *Therion*.

- **2006**

Octobre : *Therion* 0.4 — Nouveau visualiseur 3D (Loch).

- **2007**

Février : *Therion* 0.5 — Prise en charge du morphing des esquisses bitmap.

### Le futur

Bien que *Therion* soit déjà utilisé pour la production de topographies, de nombreuses nouvelles fonctionnalités devraient être implémentées à l'avenir :

#### **Général**

- informations de fermeture de boucle en SQL.

#### **Dessins 2D**

- compléter les jeux de symboles prédéfinis
- générer des registres pour l'atlas
- utiliser MPlib au lieu de METAPOST

#### **Modèles 3D**

- améliorer la modélisation des parois de galerie et salles

#### **XTherion**

- améliorer les capacités de montage 2D

#### **Loch**

- schémas de couleurs.
- arborescence des topographies pour sélectionner les sous-topos à afficher.

- filtrage spatial (par exemple, découpage en plans).
- support pour plusieurs surfaces.

### **Labyrinth**

- nouvelle interface graphique dans un futur lointain (voir <https://labyrinth.speleo.sk>)

*Μὴ εἶναι βασιλικὴν ἀτραπὸν ἐπὶ γεωμετρίαν.*

*Il n'y a pas de voie royale vers la géométrie.*

*\* Euclide, 3rd century BC \**

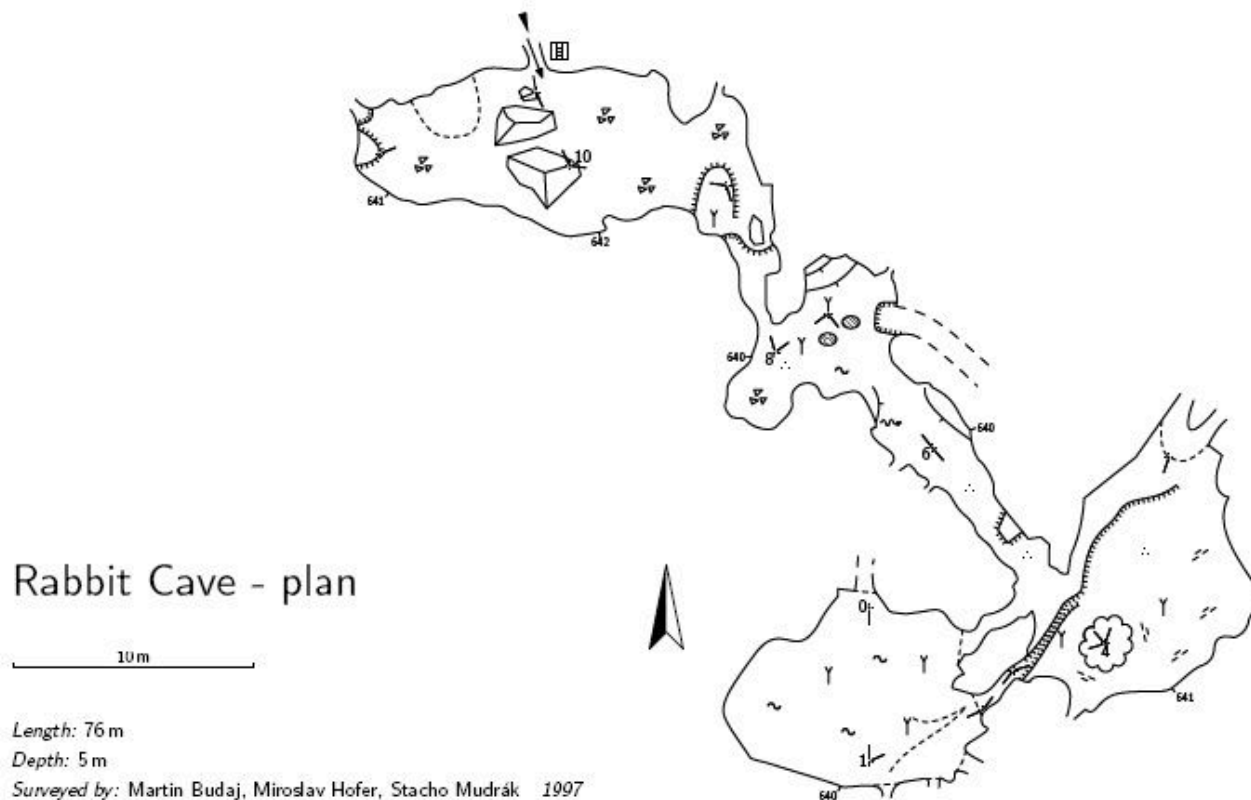
*Il n'y a pas de route royale, mais il y a une route.*

*\* Frederick P. Brooks, 1987 \**

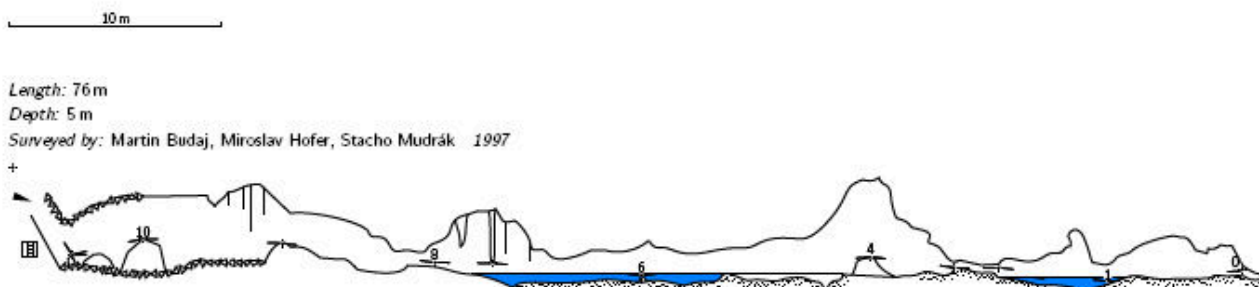
## Études de cas

Ce chapitre est automatiquement généré à partir du répertoire 'samples' du code source de *Therion*.

### Dessiner une topographie avec *Therion*



### Rabbit Cave – extended elevation



### Listing de cavit es

Lorsque vous explorez des cavit es dans une zone  tendue, vous devez souvent g n rer une liste   jour de toutes les grottes de cette r gion. Dans le jeu de donn es *Therion*, le terme «cave» peut  tre d fini de diff rentes mani res, en fonction du niveau de relev  topographique de ces cavit es.

Si vous venez de relever le point d'entrée (par exemple, en utilisant un GPS ou des méthodes de pointage standard), il vous suffit d'attribuer le marqueur `entrance` à l'aide des éléments suivants :

```
station cave-1 "Grotte non topographiée" entrance
```

Si une galerie fait suite à cette entrée, mais que vous n'en avez pas terminé l'exploration, vous devez également ajouter le marqueur `continuation` et le marqueur `explored`, suivis de la longueur explorée. Les longueurs de ces parties explorées mais non mesurées font également partie des statistiques récapitulatives.

```
station cave-2 "Non topo mais explorée" entrance \ continuation explored 80m
```

Si vous avez correctement exploré une cavité, ses données sont généralement dans une structure de relevé topographique (`survey`). Le levé devient une cavité lorsque sa station topo d'entrée principale est spécifiée à l'aide de l'option `-entrance`. Exemple:

```
survey small_cave -title "Grotte Tunnel" -entrance 0
```

Si le relevé contient des données topo, mais qu'aucune entrée principale n'est spécifiée, il n'est pas considéré comme une cavité, mais simplement comme unité d'organisation des données et n'est donc pas présent dans la liste des cavités. Seuls les relevés contenant certaines cavités (par exemple : station d'entrée ou relevés avec entrée principale définie) sont présents dans le tableau final de la liste des grottes.

Une fois que les cavités ont été correctement définies dans vos données, vous pouvez générer un tableau à l'aide de la commande :

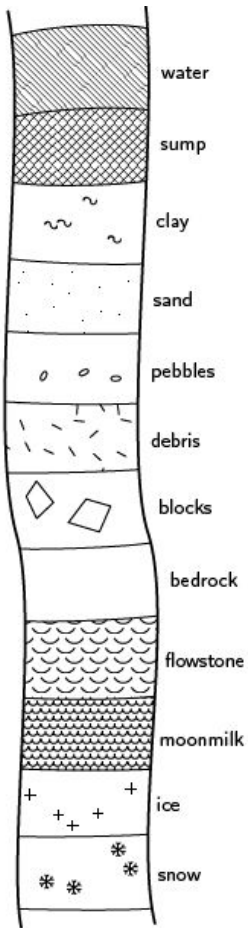
```
export cave-list -o caves.html
```

Le tableau correspondant ressemble à ça :

Titre	Long.	Prof.	Exploré	Altitude
Grotte Tunnel	25	7		1244.0
Grotte non topographiée				1234.0
Non topo mais explorée			80	1256.0

## Symboles de surfaces

### *Types de surfaces Therion*



- eau
- siphon
- argile
- sable
- cailloux
- débris
- blocs
- roche en place (substrat rocheux)
- coulée de calcite
- moonmilk
- glace
- neige

**Morphing de croquis**

Esquisse de la topographie originale :

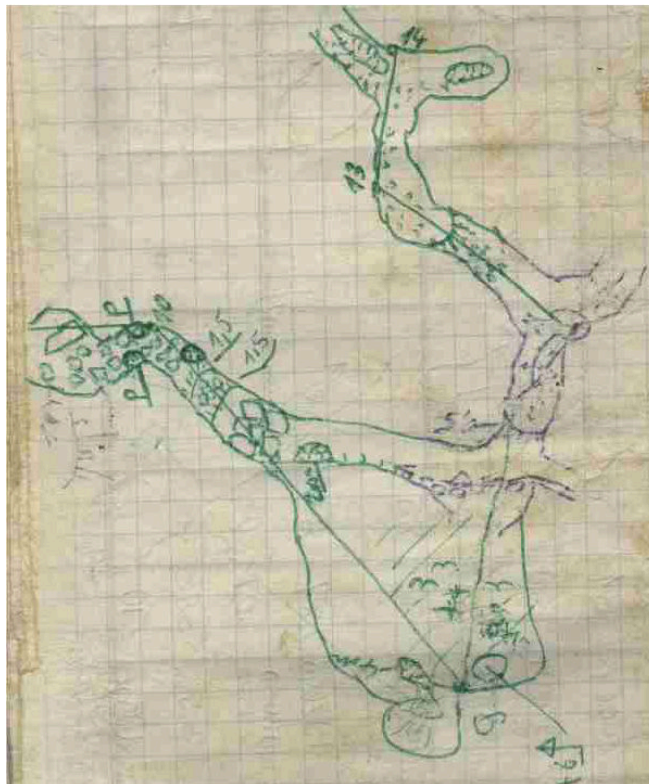
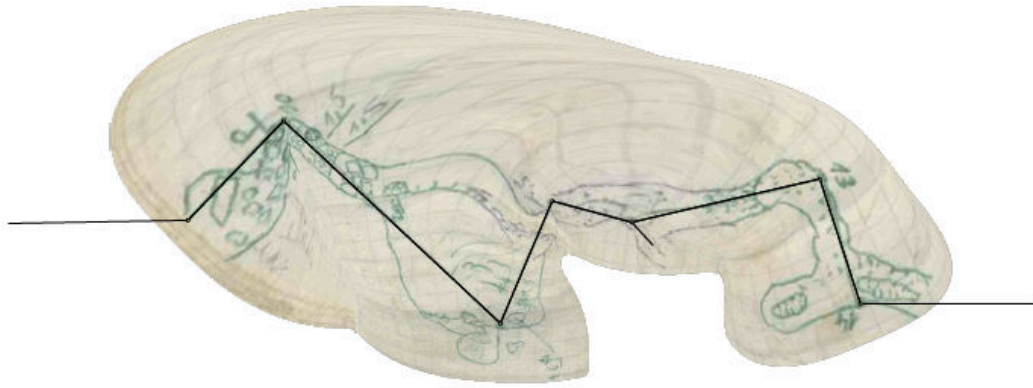
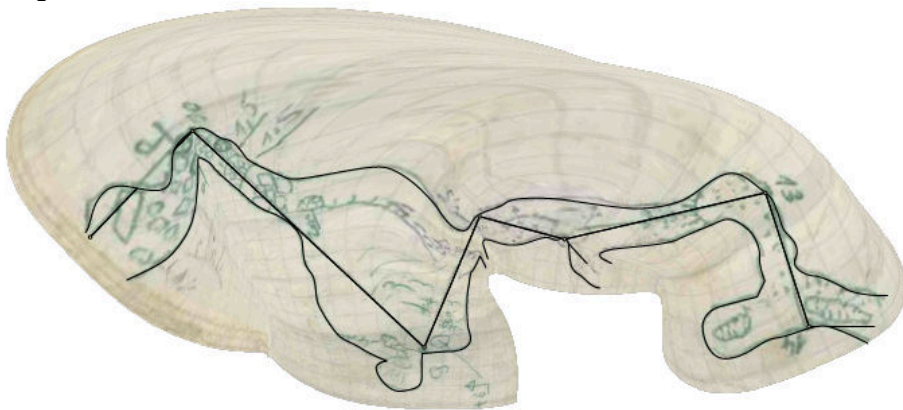


Image après morphing :





Parois après morphing :



Une autre esquisse de topographie :

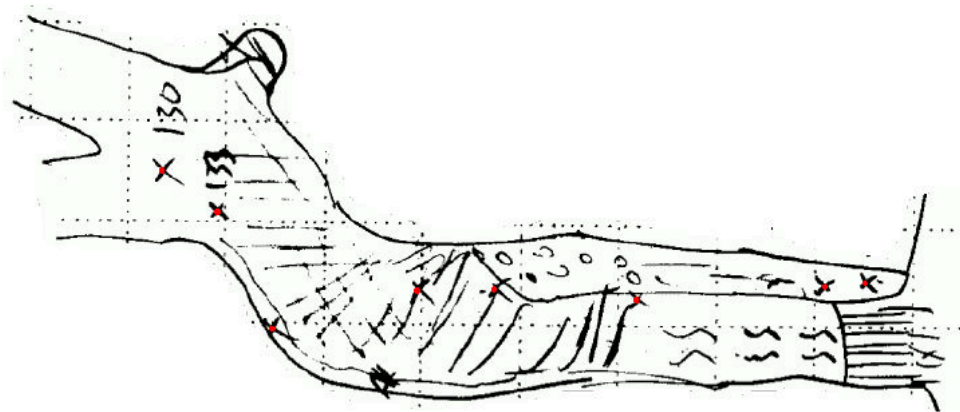


Image après morphing :

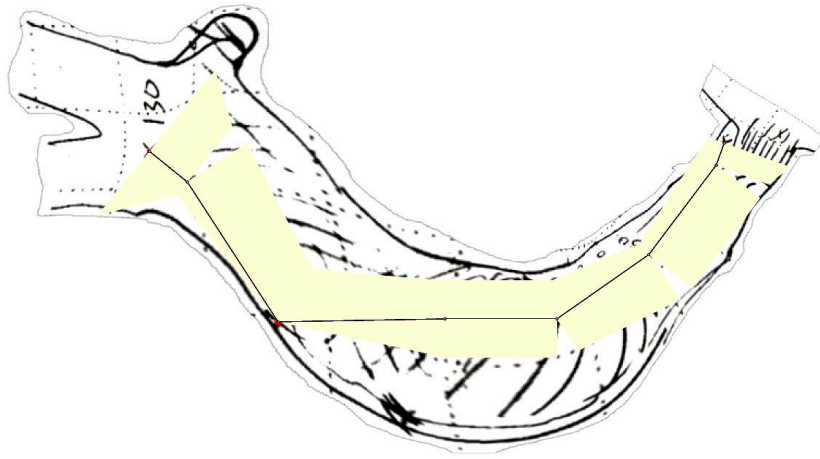
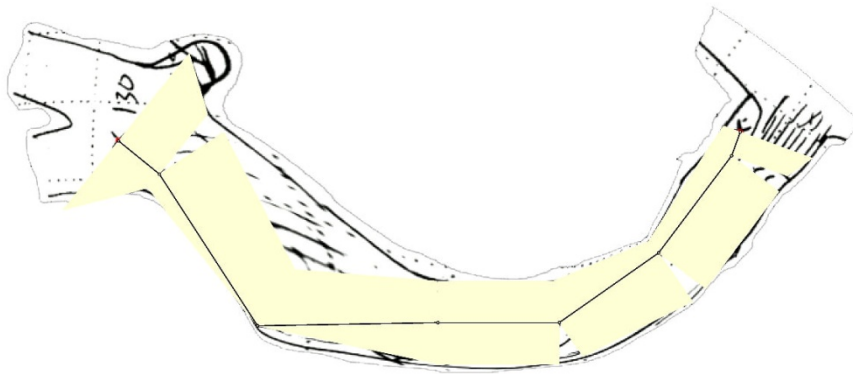
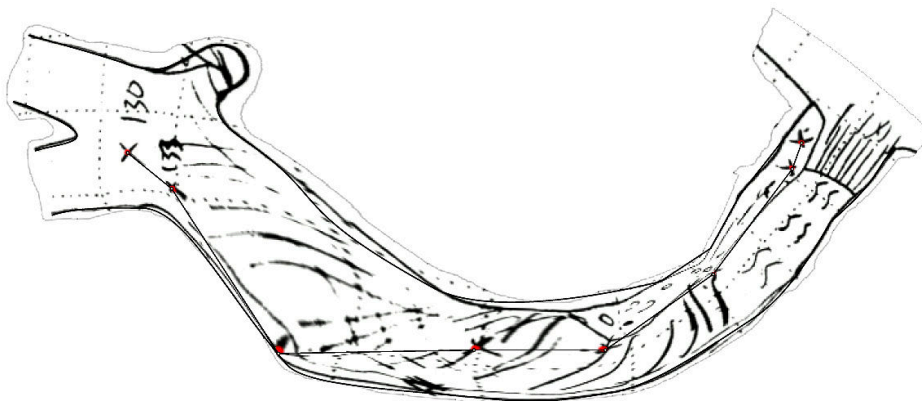


Image après morphing et insertion de points :

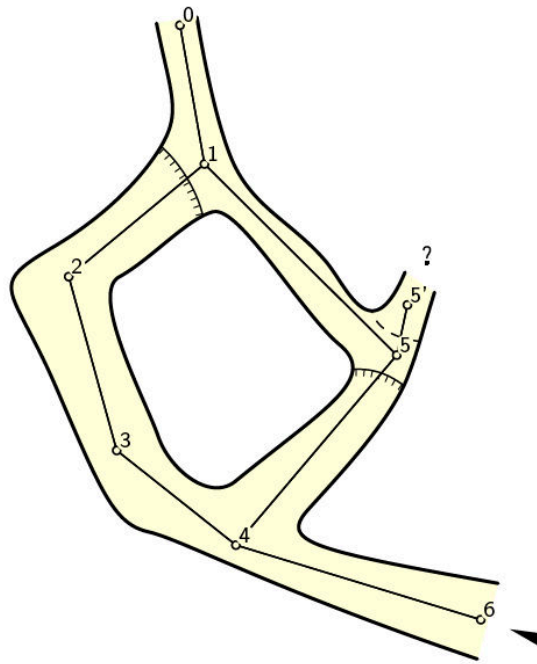


Et avec morphing des parois :

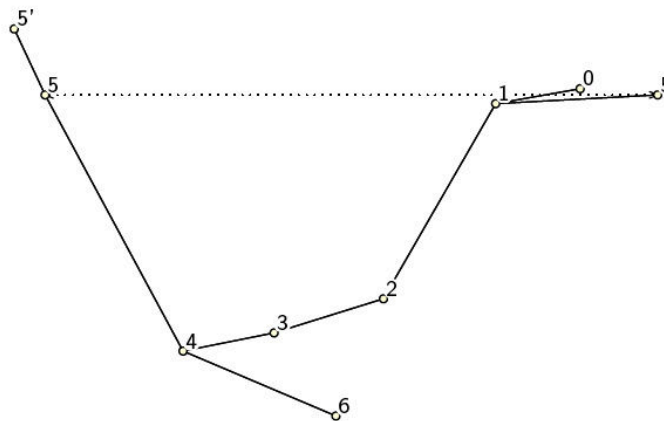


**Contrôle de coupe développée**

Supposons que la situation suivante (vue de dessus) représente une boucle entre les stations 1 et 4, une petite cheminée à proximité de la station 5 et l'entrée de la grotte à la station 6.



Par défaut, la coupe développée de la ligne de cheminement de cette situation ressemble à ceci :



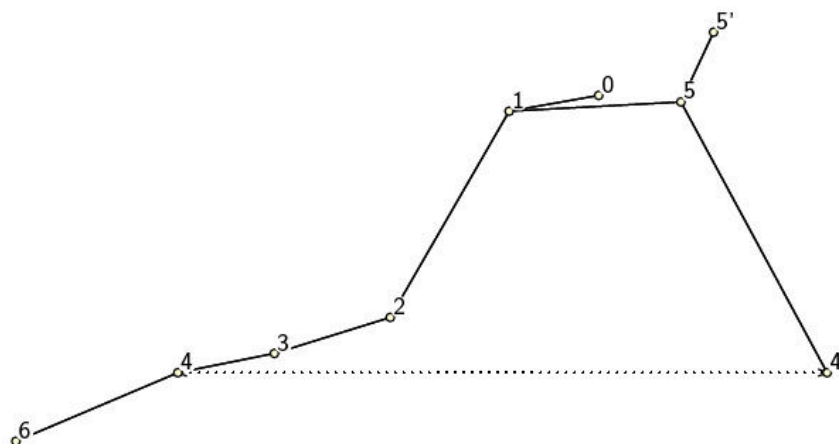
Ce n'est évidemment pas ce que nous aimerions obtenir.

Pour contrôler le processus coupe développée, il existe une option d'extension spéciale dans la commande `centerline`.

Tout d'abord, nous aimerions commencer notre coupe développée à la station 6 (où se trouve l'entrée). Cela peut se faire en spécifiant sur la ligne de cheminement :

```
extend start 6
```

C'est déjà mieux !



Mais il y a toujours un problème avec une branche contenant la station 5.

Si nous souhaitons démarrer cette branche à partir de la station 4, nous devons interdire à *Therion* d'étendre la station 5 à partir de 1. Cela peut se faire en spécifiant :

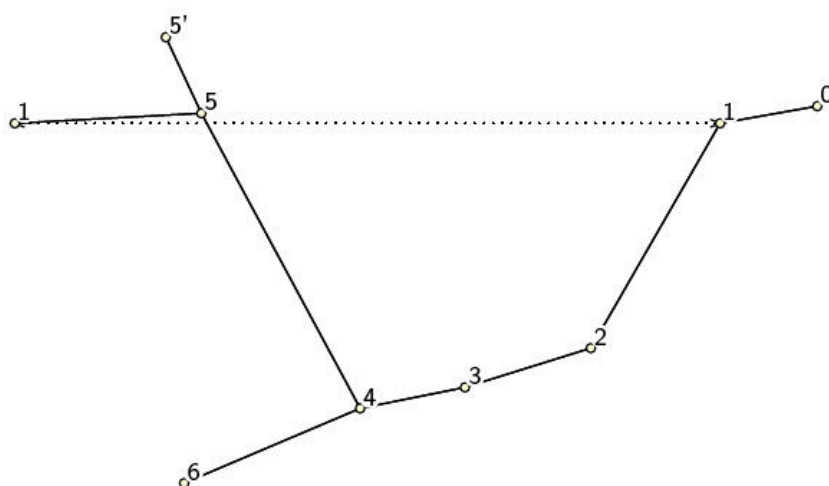
```
extend ignore 1 5
```

Cela signifie que cette visée de 1 à 5 sera ignorée lors de la génération d'une coupe développée.

Si nous souhaitons étendre la branche commençant par la station 5 du côté gauche, nous devons également spécifier :

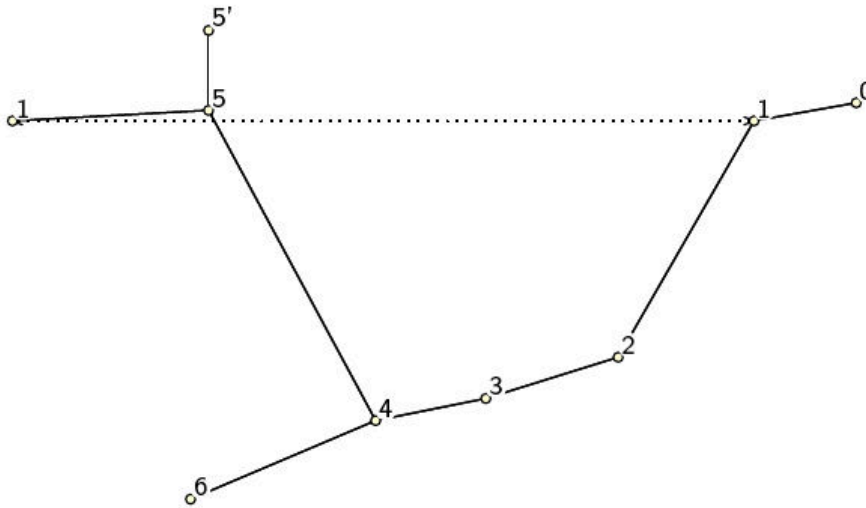
```
extend left 5
```

Cela étendra toutes les stations de la station 5 vers la gauche.



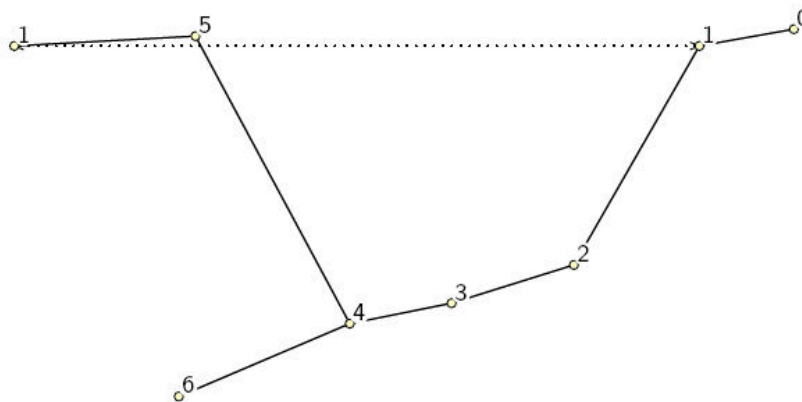
Comme nous l'avons mentionné précédemment, il y a une petite cheminée au-dessus de la station 5. Dans ce cas, il est beaucoup plus naturel de dessiner cette prise de vue à la verticale et non inclinée (car il s'agit d'une cheminée). Pour spécifier ceci, utilisez :

```
extend vertical 5 5'
```



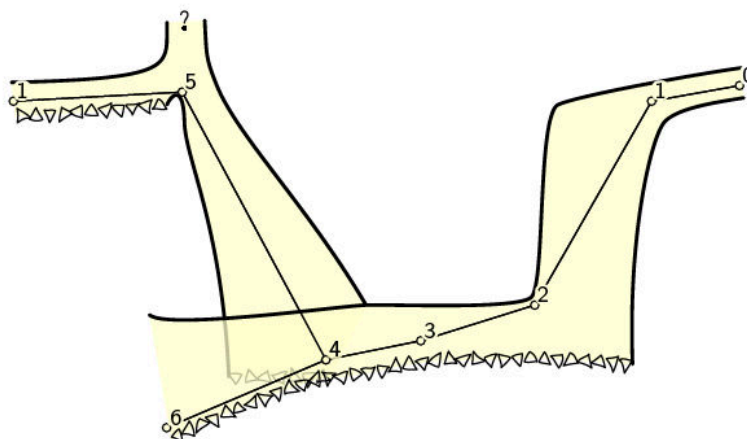
Ou vous pouvez aussi cacher complètement cette partie de la ligne médiane étendue en utilisant :

```
extend hide 5 5'
```



**Les stations sur des segments en coupe développée**

Même si la station 1 est présente plus d'une fois sur la carte, *Therion* détecte automatiquement la position correcte de cette station dans chaque segment et elles sont généralement dessinées correctement.

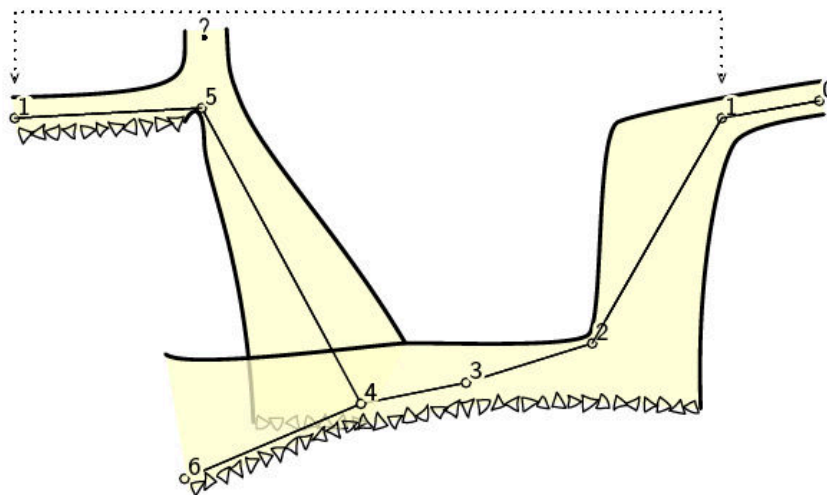


La seule chose qui manque est la ligne de connexion entre les stations 1 de ces deux branches. *Therion* ne génère pas automatiquement ces lignes car leur forme dépend généralement de chaque topographie particulière.

Pour tracer une telle ligne, il vous suffit de créer un segment de calque (*scrap*) simple avec cette ligne. Voici un exemple :

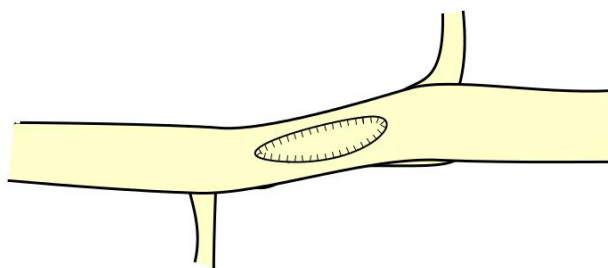
```
scrap sc -proj extended
  point 0 0 station -name 1 -from 5 -visibility off point
  100 0 station -name 1 -from 2 -visibility off
  line map-connection
    0 5
    0 15
    100 15
    100 5
  endline
endscrap
```

Comme vous pouvez le constater, même si deux stations portent le même nom, elles sont distinguées par l'option *-from*, qui spécifie la station précédente en coupe développée. En utilisant ce segment de calque (*scrap*), vous obtiendrez le dessin final suivant :



### Affichage décalé de recouvrement de topos.

Supposez qu'il y a deux topos *m1* et *m2* (*m1* situé au-dessus de *m2*). Par défaut *Therion* montre *m1* recouvrant *m2*.



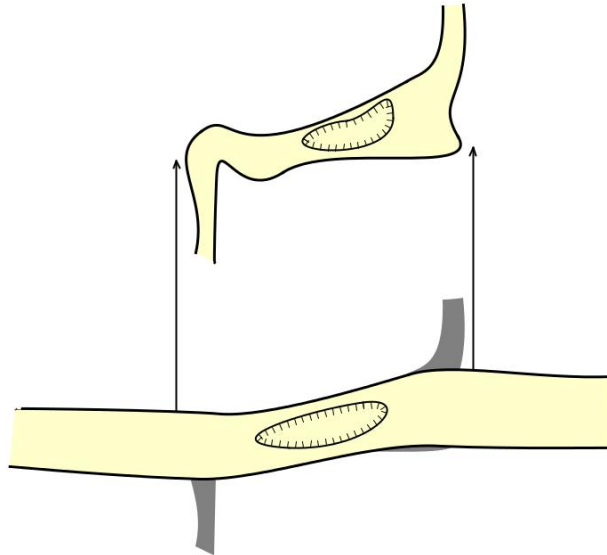
Pour afficher la topographie *m2* en décalé, vous devez créer une topographie (*map*) contenant *m2* et spécifier le type de décalage et de prévisualisation pour *m2* sur cette topo. Par exemple appelons-la *m12* :

```

map m12
  m1
  break
  m2 [0 8 m] below
endmap

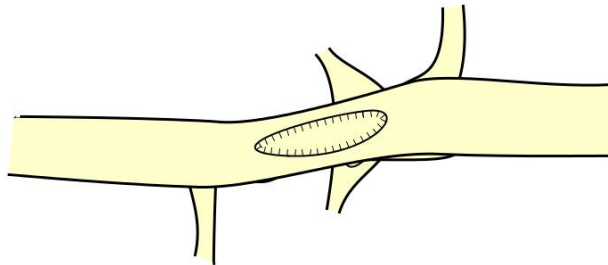
```

Après avoir sélectionné le fichier *m12* dans le fichier de configuration, vous obtiendrez :



Les flèches de jonction sont créées à partir des points de connexion spécifiés (*map-connection*) sur la topographie dans les segments qui sont déplacés. Sinon, ces points sont ignorés.

Si la situation est plus complexe, par exemple si *m3* se situe sous *m2*...



...vous devez créer une structure plus complexe pour gérer cela.

```

map m23
  m2
  break
  m3 [-8 0 m] below
endmap

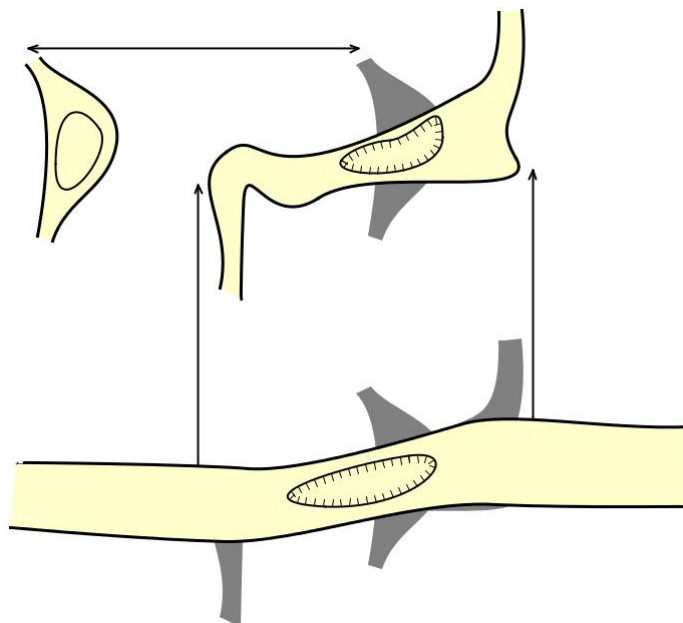
```

```

map m123
  m1
  break
  m23 [0 8 m] below
endmap

```

Alors *m123* ressemblera à ceci :



### Importer des fichiers *Survex* .3d

Les relevés *Therion* correspondent presque aux préfixes de station définis par le couple `*begin/*end` dans les fichiers source de *Survex*. Généralement, il n'existe pas de mappage un à un entre eux. Donc, si vous voulez conserver la ligne de cheminement qui se trouve dans vos fichiers *Survex*, vous devez résoudre le problème de la correspondance des structures de données *Survex* et *Therion*. Cet exemple montre trois manières différentes de traiter les préfixes de station lors de l'importation de fichiers *Survex* .3d vers *Therion*.

#### **Utilisation des topographies spécifiées dans les fichiers .th**

Si vous importez un fichier .3d avec le commutateur `-surveys use`, alors *Therion* essaie de trouver la correspondance entre le préfixe *Survex* et le nom sur le relevé topo *Therion*. Si cette correspondance est trouvée, les stations sont insérées dans le relevé trouvé. Sinon, le préfixe est laissé avec les noms de station. Exemple de code :

```
import use.3d -surveys use
input use-out.th2
survey use
  input use-in.th2
endsurvey
```

Dans ce cas, vous devez faire attention à l'endroit où vous entrez vos fichiers .th2 contenant des segments de calque (*scraps*). Dans le fichier `use-out.th2`, vous devez vous référer aux noms de stations en utilisant :

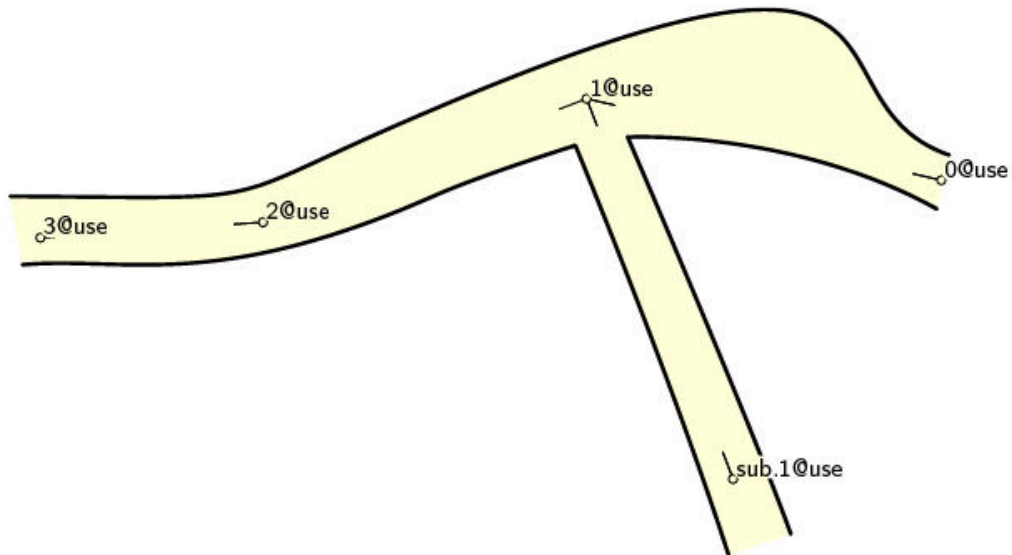
```
point 165.744 176.58 station -name 2@use
```

mais dans le fichier `use-in.th` en utilisant

```
point 321.454 236.22 station -name 1
```

La topographie finale avec les noms de station dans ce cas ressemble à ceci :





Les noms de station sont avant le symbole "@", les noms de relevé suivent ce symbole.

### **Création de topographies inexistantes**

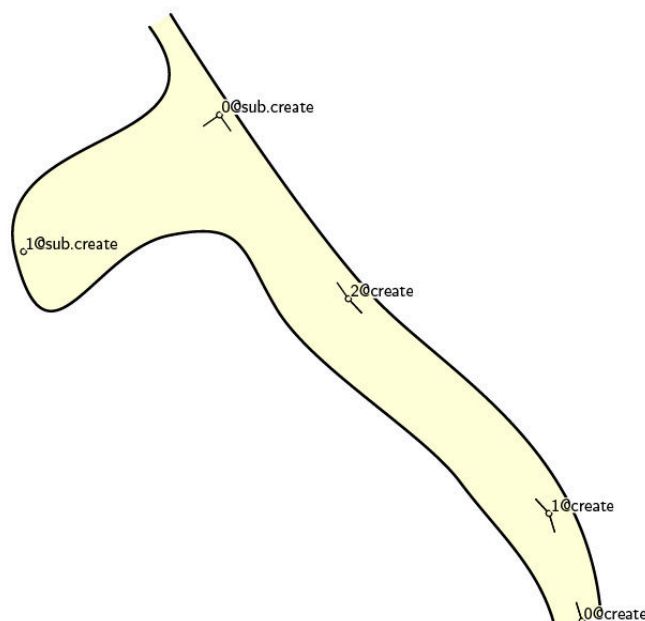
Si vous importez votre fichier .3d en utilisant :

```
import create.3d -surveys create
```

tous les préfixes *Survex* sont pris en compte et, s'il n'existe pas de topos portant les noms correspondants, elles sont créées. Dans ce cas, si vous faites :

```
input create.th2
```

juste après la commande d'importation (`import`), vous vous référez aux stations du fichier .th2 dont les noms sont indiqués sur la topographie.

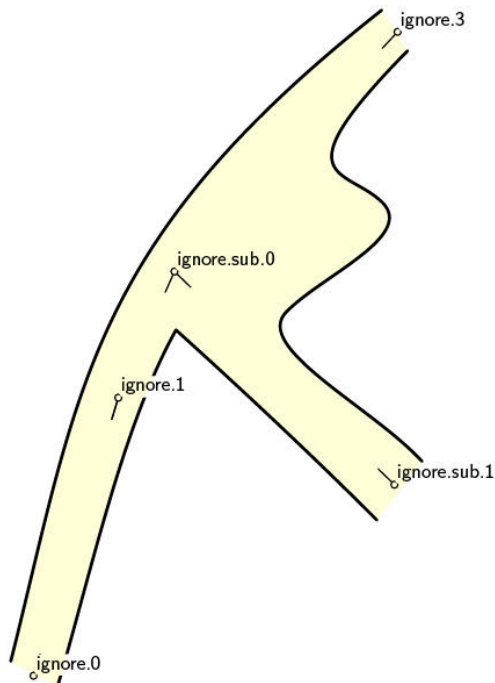


## Ignorer les préfixes de stations

La dernière possibilité est d'importer votre fichier .3d en utilisant :

```
import ignore.3d -surveys ignore
```

Dans ce cas également, les références de station du fichier .th2 correspondant sont identiques aux noms de station de la topographie finale suivante.



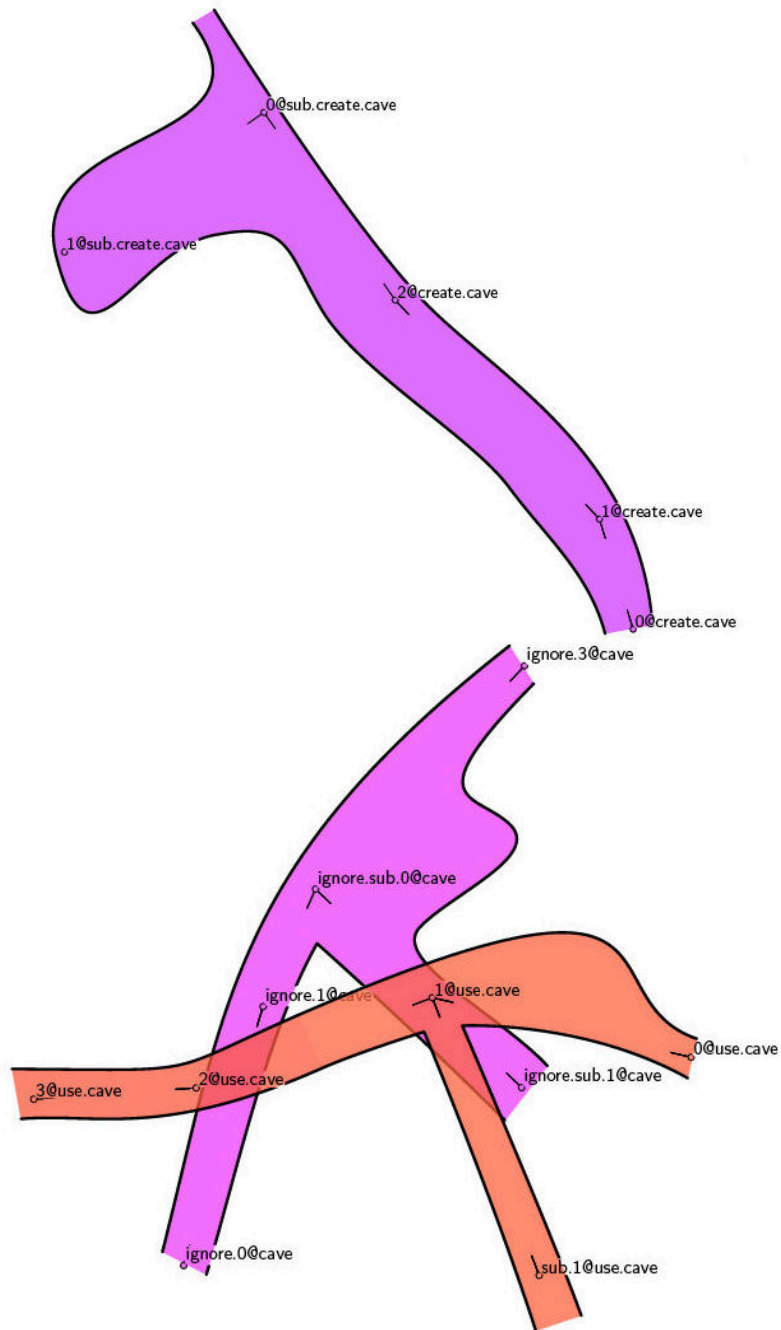
Notez qu'aucune nouvelle topographie n'est créée dans ce cas. Les noms de station sont extraits des fichiers .3d sans modification.

## Gérer des projets complexes

Supposons une situation dans laquelle vous souhaitez réunir ces trois petites topographies dans un seul grand projet. Supposons que les coordonnées des entrées de la grotte soient spécifiées dans le fichier cave.3d de niveau supérieur. Si votre code de fusion est celui-ci :

```
import cave.3d -surveys use
survey cave
  input use/use.th
  input create/create.th
  input ignore/ignore.th
endsurvey
```

Alors toutes les stations ne seront pas remplacées correctement. La série « créée » est mal placée sur la topographie finale.

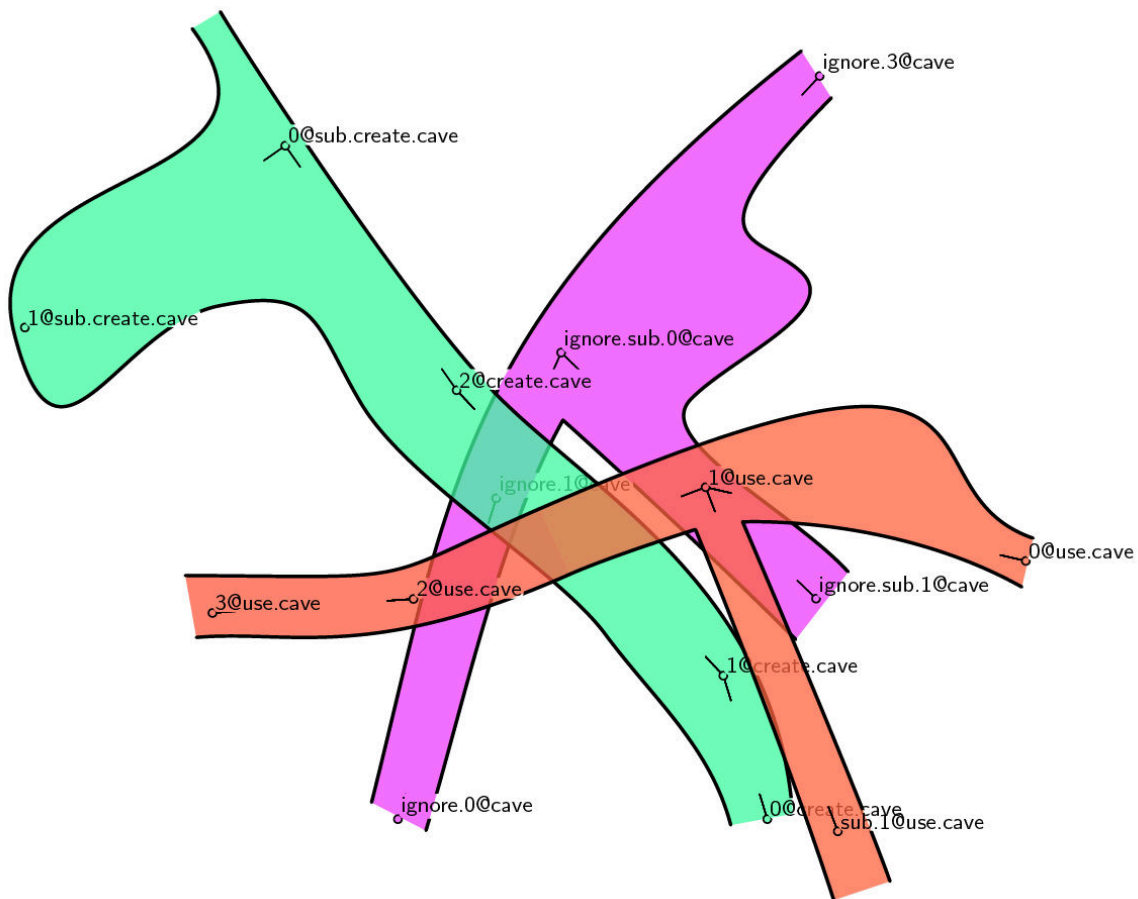


En effet, dans la commande d'importation de niveau supérieur, 'survey create' est importé depuis le fichier `create/create.3d` avec des coordonnées incorrectes et nous importons le fichier `.3d` de niveau supérieur avec le commutateur `-surveys use`. Cela signifie que 'cave.create.1' sera importé ici en tant que 'create.1@cave' et non '1@create.cave'.

Pour résoudre ce problème, nous devons réimporter les stations du fichier `cave.3d` de niveau supérieur une fois de plus avec le commutateur `-surveys create`.

```
import cave.3d -surveys use
import cave.3d -surveys create
survey cave
  input use/use.th
  input create/create.th
  input ignore/ignore.th
endsurvey
```

Après cette dernière importation, l'apparence finale obtenue sera celle-ci.



## Conclusion

Même s'il existe plusieurs possibilités pour mapper la structure des préfixes *Survex* à la structure du relevé *Therion*, la solution la plus propre consiste à créer une structure de relevé ayant la profondeur souhaitée dans les fichiers `.th` à l'aide de couples `survey/endsurvey` vides et de toujours utiliser `-surveys use` en important des dossiers `.3d`.

## Gestion des points d'interrogation

Les suites possibles sont traitées de manière spéciale dans *Therion* à la fois dans les fichiers de la ligne de cheminement et de la topographie finale. Vous pouvez associer une description textuelle, une longueur explorée (derrière cette suite) ou tout autre attribut (par exemple, `Code` si vous avez votre propre norme de codage pour les continuations).

### Indicateurs de points d'interrogation (continuation) sur la ligne de cheminement

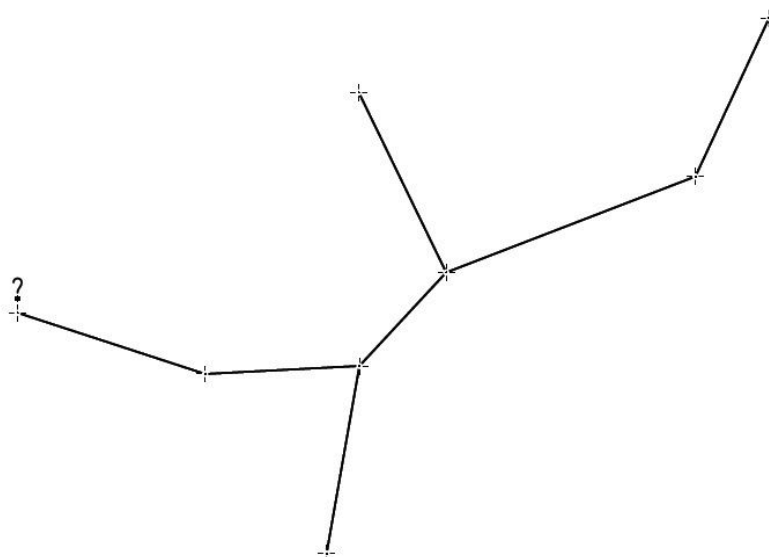
Dans l'objet `centerline`, vous pouvez ajouter un indicateur spécial à la station, où une continuation de l'exploration semble possible. Il suffit d'utiliser la syntaxe suivante

```
station 5 "pit" continuation attr Code V explored 20m
```

Lorsque vous exportez la topographie finale et utilisez

```
symbol-show point flag:continuation
```

dans votre mise en page (`layout`), la station avec l'indicateur de continuation spécifié est marquée par un point d'interrogation. Le symbole de continuation s'affiche au-dessus de la station (voir ci-dessous).



Vous pouvez redéfinir le symbole de continuation pour afficher également la description de la suite (stockée dans l'attribut `_text`) à l'aide du code de présentation suivant :

```
code metapost
  def p_continuation(expr pos,theta,sc,al) =

    % draw default continuation symbol
    p_continuation_UIS(pos,theta,sc,al);

    % if text attribute is set
    if known(ATTR__text) and picture(ATTR__text):

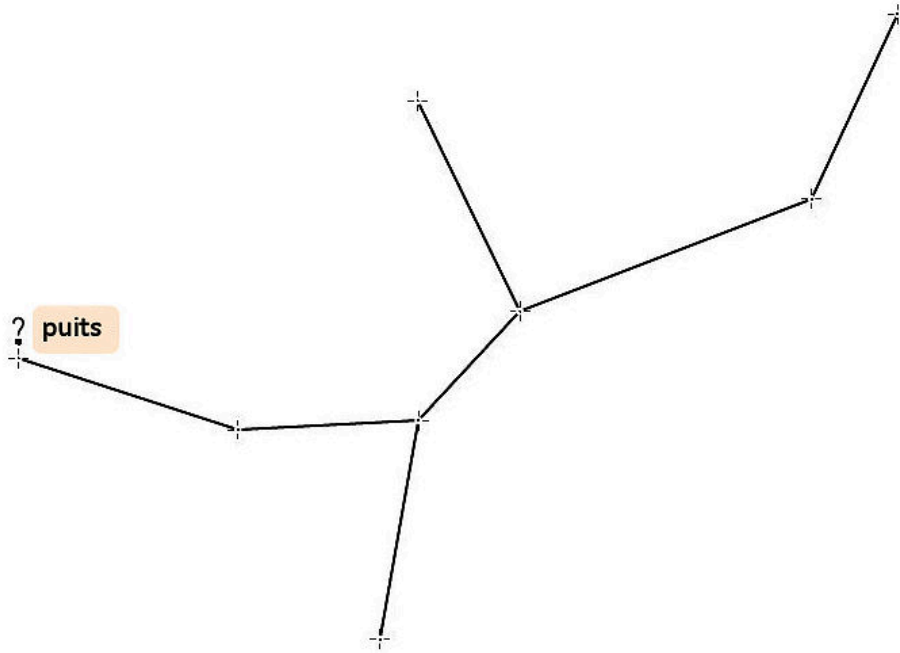
      % set labeling color to light orange
      push_label_fill_color(1.0, 0.9, 0.8);

      % draw filled label with text next to ?
      p_label.urt(ATTR__text,(.5u,-.25u) transformed T,0.0,8);

      % restore original labeling color
      pop_label_fill_color;

    fi;
  enddef;
endcode
```

La description de la suite (continuation), si elle existe, est également affichée sur la topographie finale.

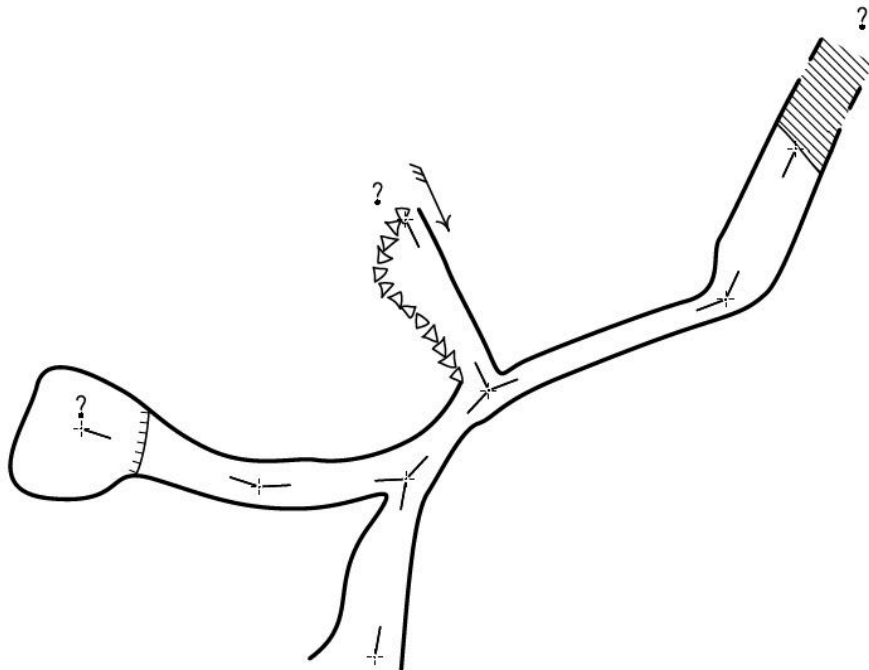


**Indicateurs de points d'interrogation (continuation) sur le dessin**

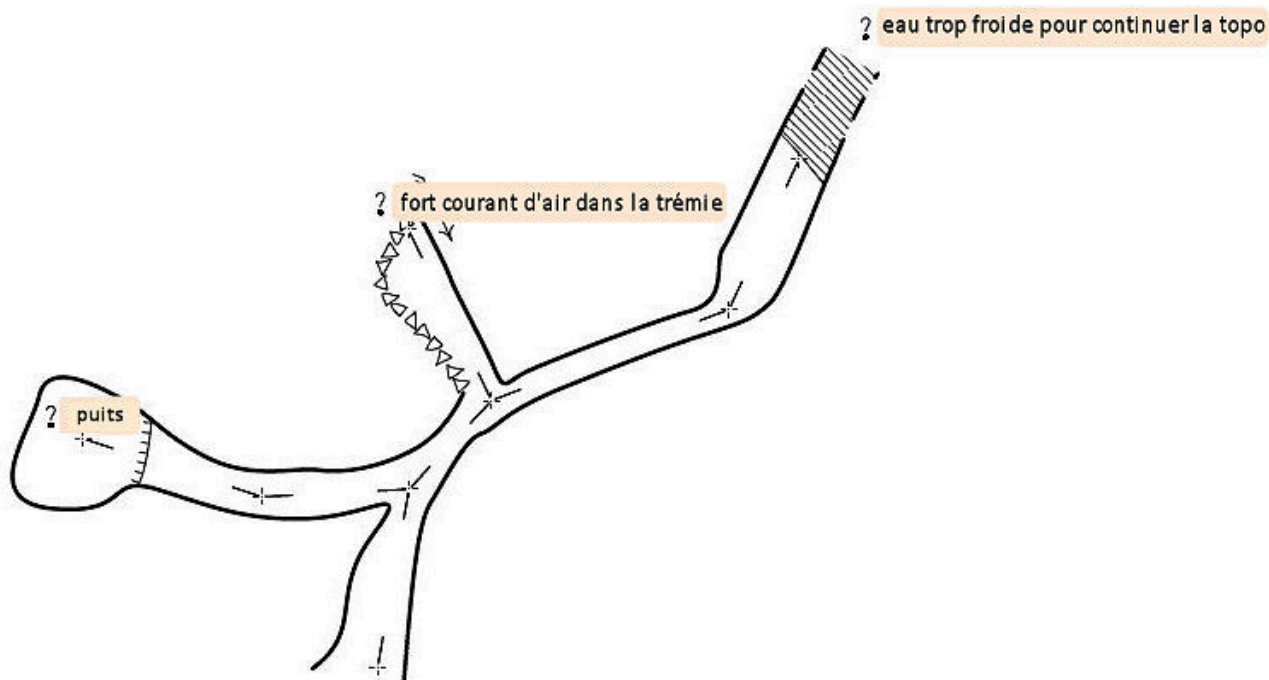
Sur le segment (.th2 file), vous pouvez utiliser les options `-text`, `-code` et `-explored` pour décrire le symbole de suite de galerie (continuation).

```
point 796.0 676.0 continuation -attr Code A -explored 50m \
  -text "eau trop froide pour continuer la topo"
```

Sur la topographie finale, seuls les points d'interrogation sont affichés par défaut.



Mais lorsque vous redéfinissez le symbole de continuation comme indiqué ci-dessus, vous pouvez également afficher les codes et les descriptions de continuité (voir figure suivante).



### Exportation de listes de marqueurs de points d'interrogation

Vous pouvez également exporter une liste de tous les marqueurs « ? » de votre relevé en utilisant `export continuation-list -o questions.html`

Le fichier `questions.html` contient alors la liste suivante :

Comment	Explored	Survey	Station	Code
eau trop froide pour continuer la topo	50.0	Sample cave	7	A
fort courant d'air dans la trémie		Sample cave	3	B
puits	20.0	Sample cave	5	V

### Utiliser des types de symboles définis par l'utilisateur

Si *Therion* n'offre pas de symbole approprié pour l'entité que vous voulez dessiner sur la topo, vous pouvez utiliser un type de symbole défini par l'utilisateur (type `u`, valide pour les objets de type point, ligne et surface).

La syntaxe est très simple. Supposons que vous souhaitiez créer un point, une ligne et une zone «chauve-souris». Vous utilisez simplement `u:bat` (ou `u:chiro`) comme type (`bat` ou `chiro` est en fait un sous-type) de type `u`. Donc, votre code ressemblera à ça :

```
point 555.0 480.0 u:bat (ou point 555.0 480.0 u:chiro etc.)
```

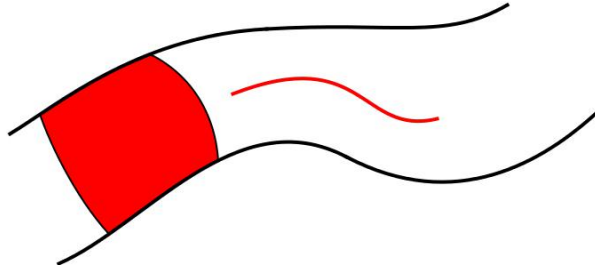
ou

```
line u:bat
```

ou

```
area u:bat
```

Lorsque vous exportez une topographie sans définir les symboles dans METAPOST, les symboles définis par l'utilisateur sont mis en surbrillance en rouge, sans aucune représentation graphique.



Pour les afficher correctement, vous devez définir leurs symboles dans le langage METAPOST de la même manière que les symboles ordinaires sont généralement redéfinis.

Tout d'abord le symbole du point. Dans le code METAPOST la section de votre mise en page qui définit le point `u:bat symbol` se présente comme ceci :

```
def p_u_bat(expr pos, theta, sc, al) =  
  T := identity shifted pos;  
  thfill (bat_path scaled 2.0)  
enddef;
```

même chose dans la ligne `u:bat symbol`

```
def l_u_bat(expr P) =  
  T :=identity;  
  cas := 0;  
  dlzka := arclength P;  
  mojkkrok:=adjust_step(dlzka, 1.0u);  
  pickup PenD;  
  forever:  
    t := arctime cas of P;  
    thfill bat_path scaled 0.5 shifted (point t of P)  
    cas := cas + mojkkrok;  
    exitif cas > dlzka + (mojkkrok / 3); % for rounding errors  
  endfor;  
enddef;
```

et finalement la zone `u:bat symbol` (motif dans ce cas)

```
% bat pattern  
beginpattern(pattern_bat);  
fill bat_path withcolor black;  
endpattern;  
  
% bat area symbol  
def a_u_bat (expr Path) =  
  T :=identity;  
  thclean Path;  
  thfill Path withpattern  
pattern_bat;  
enddef;
```



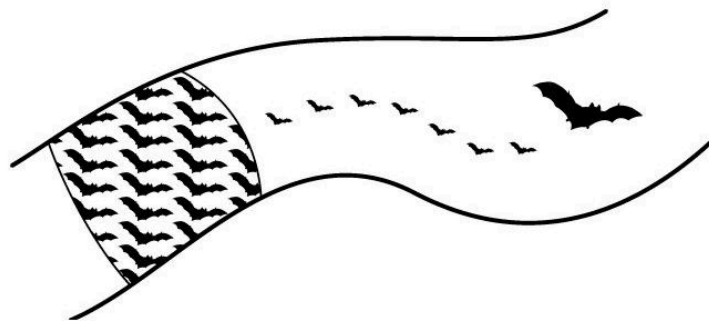
Ces symboles seront également inclus dans la légende. Pour changer la façon dont ils sont dessinés, définissez simplement la macro appropriée. Son nom doit être le nom de la macro du symbole suivi du suffixe `_legend`.

```
def l_u_bat_legend =  
  l_u_bat(((.2,.2) -- (.8,.8)) inscale)  
enddef;
```

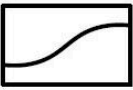

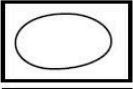


Enfin, ajoutez une description de vos nouveaux symboles qui figureront dans la légende à l'aide de la commande `text` partout dans le fichier de configuration.

```
text en "point u:bat" "chiroptère"  
text en "line u:bat" "passage de ch."  
text en "area u:bat" "colonie de ch."
```

Après toutes ces définitions, vous obtiendrez des symboles de point, de ligne et de surface spécifiques pour les chiroptères avec la représentation graphique appropriée ainsi que des zones de légende.



### Légende

	paroi		passage de ch.
	limite		colonie de ch.
	chiroptère		

# LA TOPOGRAPHIE AVEC THERION

*par Xavier Robert  
du Groupe Spéléologique Vulcain*



# 1. PRÉAMBULE

Encore beaucoup de spéléos d'exploration font leurs reports topographiques à l'aide de deux logiciels distincts : le premier (comme le très pratique et convivial [Visual Topo](#)) permet de construire le squelette de la topographie et de l'exporter dans un format vectoriel, qui est ensuite importé dans le second logiciel, vectoriel, dit de DAO (comme [Inkscape](#) ou Adobe Illustrator), permettant de dessiner autour du squelette. Le logiciel Visual Topo est simple d'utilisation. Le logiciel de DAO demande quant à lui un petit temps de formation pour pouvoir l'utiliser correctement. Cette méthode est adaptée aux cavités simples et à la publication de topographie sans intégration dans un Système d'Information Géographique (SIG).

En revanche, à partir du moment où nous travaillons sur des cavités complexes avec de nombreux bouclages ou sur des systèmes karstiques avec plusieurs cavités, cette méthode devient difficile, voir très difficile à mettre en œuvre. En effet, ici, chaque dessin est fixe et indépendant de la donnée topographique. Cela pose problème pour connecter deux cavités ou pour ajouter un nouveau réseau/bouclage dans une cavité connue, parce que ce bouclage déforme le squelette, et au bout d'un certain nombre de bouclages, nous n'arrivons plus à relier les dessins de façon acceptable.

C'est pourquoi, des logiciels spécifiques, intégrant à la fois le travail de base sur les données, et le dessin, qui est alors ancré sur le squelette, sont développés depuis quelques années. L'avantage, c'est qu'à chaque nouvel ajout, l'ancien dessin est déformé automatiquement pour coller au nouveau squelette topographique. Cela permet donc l'évolution de la topographie d'une cavité complexe ou d'un système au cours du temps. En France, trois tels logiciels sont connus : [GHTopo](#), [TopoCalc'R](#) et [Therion](#). *Therion* étant plus ancien et multi-OS (contrairement aux deux autres), nous avons commencé à nous y intéresser avant le développement de *TopoCalc'R*, et nous y sommes restés. Son avantage est que non seulement il permet ce travail sur la topographie d'une cavité complexe, mais aussi, il permet de travailler de façon robuste à l'échelle d'un système karstique complet.

*Therion*... Arf, rien ne sert de s'enfuir ! Tout le monde a pour idée que c'est un truc de Geeks, que c'est imbitable et difficile à prendre en main. Mais pourtant, ce n'est pas si compliqué que ça, du moment que nous prenons le temps de nous y former (et ce n'est pas plus difficile que de se former correctement à un logiciel de dessin vectoriel) et de travailler de façon rigoureuse. Effectivement, si vous êtes allergique à écrire 3 mots consécutifs ou à lire un fichier texte, alors, utiliser *Therion* risque d'être difficile. Mais si vous avez déjà écrit un SMS, un e-mail, ou un petit document Word sans mise en page, alors, il n'y a aucune raison que vous n'y arriviez pas, sauf à faire un blocage psychologique ou à y mettre de la mauvaise volonté...

Ces pages ne remplacent pas la pratique et l'expérience, mais elles apportent une base pour l'utilisation rigoureuse de *Therion*. Ce n'est évidemment pas un tutoriel exhaustif, mais vous trouverez des notions de base, mais aussi des fiches rappels synthétiques qu'il peut être bon de garder sous la main. Cela peut aussi servir à améliorer votre prise de notes sous terre pour pouvoir produire des données qui permettent une meilleure utilisation de *Therion*.

La description des fichiers peut être rébarbative, mais il faut en passer par là pour comprendre leur fonctionnement. Par la suite, il sera possible d'utiliser des fichiers plus anciens pour les adapter aux nouvelles topographies, ce qui facilitera la tâche. Il peut être utile d'avoir sous la main des exemples de ces fichiers, afin de les comprendre, et finalement de se rendre compte, que ce n'est pas plus compliqué qu'un texte quelconque, d'autant plus qu'il y a moins de mots

## **2. SOMMAIRE**

### **1. PRÉAMBULE**

### **2. SOMMAIRE**

### **3. PREMIERS PAS**

- Installer la suite *Therion*
- Installer un « vrai » éditeur de texte (et sa coloration syntaxique)
- Des aides disponibles sur le net

### **4. QU'EST-CE THERION ? QU'EST-CE XTHERION ?**

- De quoi a besoin *Therion* ?
- Que peut produire *Therion* ?
- *Therion* ? *XTherion* ? Loch ?
- Comment utiliser le logiciel *XTherion* ?
- Quelques notions de base indispensables

### **5. ENTRER DES DONNEES TOPOGRAPHIQUES POUR THERION**

- Comment créer un fichier de données ?
- Comment ajouter une nouvelle séance topographique ?

### **6. CREER UN FICHIER DE CONFIGURATION POUR LA COMPILATION**

- Qu'est-ce qu'un fichier *.thconfig* ?
- Compiler ?
- Comment simplifier les fichiers de configuration ?

### **7. FAIRE UN DESSIN AVEC XTHERION**

- Faire un dessin avec *XTherion*
- Principaux points utilisés et leurs options classiques
- Principales lignes utilisées et leurs options classiques
- Principales aires utilisées et leurs options classiques
- Penser et gérer les calques (= scraps)
- Ajouter un nouveau dessin à la topographie existante
- Ajouter une section de galerie sur un dessin
- Spécificités des coupes développées
- Fiche synthèse des étapes à suivre pour faire un dessin

### **8. DES ERREURS CLASSIQUES**

- Comment savoir qu'il y a une erreur ?
- Des erreurs classiques
- Trouver et corriger (debugger) une erreur dans un dessin ?

### **9. PENSER « BASE DE DONNEES TOPOGRAPHIQUES » ET SIG**

- La notion de maps
- Structurer une base de données topographiques
- Homogénéiser les topographies à l'aide d'un fichier de configuration global
- Les notions de systèmes de coordonnées et de projections (*A améliorer*)
- Utiliser un logiciel SIG – principes de base
- Ajouter un Modèle Numérique de Terrain/Surface (MNT) et draper une image

## 10. PARTAGER LA BASE DE DONNEES ?

- Comment partager une base de données ?
- Quelle licence utiliser ?
- Quelques exemples...

## 11. QUELQUES EXEMPLES

## 12. TABLE DES ILLUSTRATIONS

# 3. PREMIERS PAS

- Installer la suite *Therion*

### *Télécharger le logiciel*

Pour installer la suite *Therion*, il faut télécharger le logiciel (ou les sources, en fonction de votre système d'exploitation) sur la page <https://Therion.speleo.sk/download.php>.

### *Installation sous Windows*

L'installation est simple, il suffit d'exécuter le fichier d'install, et de suivre les instructions. Toutes les dépendances nécessaires au bon fonctionnement des logiciels de la suite *Therion* sont installées automatiquement.

### *Installation sous MacOS*

Sous MacOS, l'installation est un peu plus complexe parce qu'il faut 1) installer les dépendances nécessaires au bon fonctionnement de *Therion*, et 2) compiler les sources *Therion*.

Pour cela, si ce n'est pas déjà fait, vous avez besoin d'installer :

- Xcode à partir de l'AppStore (être patient, c'est long à télécharger et installer)
- Command Line Tools (outils de commandes en ligne) : Dans Xcode aller dans Xcode → Preferences → Downloads (Téléchargement) → Command Line Tools (Outils de commandes en ligne)
- Le gestionnaire de Packages [Homebrew](https://brew.sh/). Pour cela, copier la ligne suivante dans le Terminal (voir utilisation plus bas), et taper « entrer » :

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

### *Installer les dépendances*

1. Ouvrir une fenêtre Terminal (Pour la trouver, aller dans le Launchpad, et dans la barre de recherche, taper « Terminal »)
2. Mettre à jour la base de données d'Homebrew en tapant dans le terminal :
  1. brew update
  2. brew doctor
  3. Suivre les instructions issues de brew doctor s'il y en a, et si besoin mettre à jour les applications avec la commande : brew upgrade
3. Installer les dépendances :
  1. Installer la suite MacTeX (tout ce qui touche au langage TeX, LaTeX et pdfTeX) à partir du site <https://www.tug.org/mactex/>
  2. brew install lcdf-typetools
  3. brew install wxmac

4. brew install freetype
5. brew install vtk
6. brew install --overwrite imagemagick
7. brew doctor

L'option « `--overwrite` » permet d'éviter les conflits avec les bibliothèques d'XCode qui sont simplifiées et qui ne contiennent pas toutes les bibliothèques nécessaires.

### **Compiler Therion**

1. Décompresser l'archive *Therion-5.\*.tar.gz*
2. Ouvrir une fenêtre Terminal (Pour la trouver, aller dans le Launchpad, et dans la barre de recherche, taper « Terminal » )
3. Dans le terminal, aller dans le dossier avec la commande « cd » en tapant :
  1. cd chemin/vers/le/dossier
4. taper ensuite (certaines étapes sont longues, ne pas s'inquiéter) :
  1. make config-macosx
  2. make
  3. sudo make install

### **Erreurs possibles**

Dépendamment du système utilisé, Il est possible que le démarrage de l'application *XTherion* ou Loch indique une erreur disant qu'il ne trouve pas la bibliothèque BWidget :

Error in startup script: can't find package BWidget

Dans ce cas, il faut :

Télécharger la librairie BWidget sur : <https://sourceforge.net/projects/tcllib/files/BWidget/>

Décompresser et copier la librairie BWidget dans le dossier lib/ de tcl-tk (normalement dans /usr/local/opt/tcl-tk/lib)

Modifier le fichier .profile du terminal :

1. Ouvrir une fenêtre Terminal
2. Taper « cd » puis la touche « Entrer »
3. Taper « open .profile ». Ca devrait ouvrir dans TextEdit un fichier qui s'appelle .profile (c'est normalement un fichier caché)
4. Dans le fichier .profile, copier (et peut-être adapter en fonction de votre système) les lignes
  1. export PATH="/usr/local/opt/tcl-tk/bin:\$PATH"
  2. export LDFLAGS="-L/usr/local/opt/tcl-tk/lib"
  3. export CPPFLAGS="-I/usr/local/opt/tcl-tk/include"
  4. export PKG\_CONFIG\_PATH="/usr/local/opt/tcl-tk/lib/pkgconfig"

Attention aussi, lors des mises à jour majeures du package VTK, il faut recompiler les sources, sinon, Loch ne fonctionnera plus.

### **Installation sous Linux**

Voir la [documentation en ligne](#) et le [ThBook.pdf](#).

#### **• Installer un « vrai » éditeur de texte**

*Therion* utilise différents types de fichiers pour :

- les données topographiques en elles-mêmes,
- les données sources des dessins
- les fichiers liaisons dans le cas de topographies complexes ou de systèmes karstiques
- les fichiers de configuration et de compilation

Tous ces fichiers sont des fichiers textes. Nous pouvons donc les lire, les écrire et les modifier avec n'importe quel éditeur de texte. Mais :

- sous Windows, l'éditeur de texte par défaut corrompt l'encodage des fichiers texte (i.e. ces fichiers seront difficilement utilisables par d'autres logiciels, et leur utilisation renverra des erreurs pénibles à corriger) ; Nous déconseillons très fortement de l'utiliser...
- tous ces fichiers contiennent des mots clefs spécifiques (« commandes ») de *Therion*. Un éditeur qui permet la coloration syntaxique (i.e. qui va reconnaître et colorer les mots clefs spécifiques *Therion*) va grandement faciliter la lecture et l'édition de ces fichiers.

```

1 encoding utf-8
2
3
4 # Copyright (C) 2019 Xavier Robert <xavier.robert***@***ird.fr>
5 # This work is under the Creative Commons Attribution-ShareAlike-NonCommercial License:
6 # <http://creativecommons.org/licenses/by-nc-sa/4.0/>
7
8
9 # 1-SOURCES
10 # La ligne source specifie le fichier ou sont les donnees topo
11 # (Au fichier "nomcavite.th" il faudra avoir une ligne
12 # "input "nomcavite.th2" pour specifier le fichier ou se trouvent
13 # les donnees du dessin, comme ça, ce fichier thconfig appellera
14 # nomcavite.th" et a son tour, "nomcavite.th" appellera
15 # "nomcavite.th2")
16 source CH17-tot.th
17
18 # Add entrances coordinates
19 input legendes/entrances_coordinates.th
20
21 # Add config file
22 input ../../../../config.thc
23
24 # 2-LAYOUT
25 # Debut de la definition du Layout "xviexport"
26 layout xviexport
27
28
29 # fin de la definition du layout "xviexport"
30
31 layout my_layout
32
33 # Call the config settings (Layout config inside the config.thc file)
34 copy Coords_Header_JBCHb
35 copy drawingconfig
36 copy headerl
37 copy langue-fr
38
39 # Titre
40 doc-title "CH17"
41 # Auteur
42 doc-author "Xavier Robert"
43
44 # Pour faire la topo dans le system UTM
45 # Decommenter la ligne, et remplacer xx par la zone UTM
46 cs UTM32
47
48 # "base-scale" specifie l echelle auquel nous avons

```

Exemple d'un fichier de compilation ouvert dans un éditeur de texte (ici BBEEdit) avec coloration syntaxique : les commandes sont en bleu, les commentaires en mauve, les chaînes de caractères en rouge, les nombres en orange,... Sur le côté, nous avons les numéros de lignes, et dans cet exemple là, le triangle noir ligne 27 indique que le bloc du `layout xviexport` a été rétréci pour accéder plus rapidement à la suite.

Installer un vrai éditeur de texte n'est pas obligatoire, parce que *XTherion* contient son propre éditeur de texte, mais pour l'instant, aucune coloration syntaxique n'est disponible. Voici donc quelques propositions qui peuvent être utiles.

Les utilisateurs de Windows doivent faire très attention : Sous Windows, la gestion de l'encodage des caractères et des fins de lignes n'est vraiment pas simple à faire. Si vous avez des problèmes de compilation de fichiers édités avec un éditeur externe, et que vous ne comprenez pas pourquoi la ligne indiquée comme problématique est effectivement problématique (généralement en début de fichier), c'est que très probablement votre encodage est corrompu. En ce cas, le meilleur conseil est peut-être de travailler avec l'éditeur intégré à *XTherion*.

## **Notepad++**

Sous Windows, un des meilleurs éditeurs de texte libre est le logiciel [Notepad++](#).

### **Customiser NotePad++**

Pour que NotePadd++ puisse travailler correctement avec les fichiers *Therion* :

1. Aller dans le menu « Settings »
2. Cliquer sur « Préférences »
3. Dans la boîte de dialogue qui s'ouvre, sélectionner l'onglet « New Document »
4. Activer le bouton « UTF-8 without ROM »
5. Puis cliquer sur « Apply to open ANSI file »

### **Définir le style de langage Therion pour NotePad++**

Télécharger le fichier de définition de langage

: <https://Therion.speleo.sk/wiki/media/contrib:userdefinelang-surveXTherion533-.zip> et le dézipper dans le dossier %APPDATA%\Notepad++ .

D'autres informations sont disponibles (en anglais) sur la [doc Therion en ligne](#).

### **Visual Studio Code (Microsoft) / VSCodium (Libre)**

Récemment un éditeur de texte dédié au code et gratuit a été publié par Microsoft : [Visual Studio Code](#). C'est multiplateforme (Windows, MacOS, Unix), il existe de nombreux plugins, mais le seul problème est que ce logiciel est distribué avec des trackers. Mais, le code source de ce logiciel a été mis en ligne, et un équivalent libre sans tracker a été ensuite publié : [VSCodium](#). Il est toujours multi-OS.

La force de ce logiciel, est que c'est plus qu'un simple éditeur de texte/code, par exemple :

- Il existe un plugin spécifique au langage *Therion*, [vscode-Therion-language](#), avec complétion, coloration syntaxique des mots clefs, compilation, quelques commandes utiles spécifiques *Therion*,... Pour l'installer, télécharger l'extension, puis installez là à partir de VSCodium.
- La possibilité de travailler fichier à fichier, où sous forme de projet,
- La possibilité d'interagir avec un dépôt Git,
- La présence d'un Terminal permettant notamment de compiler les *thconfig* édités !
- Et plein d'autres trucs sympas...

### **BBEdit**

La version gratuite de [BBEdit](#) fonctionne bien avec les fichiers *Therion* sous MacOS.

Xavier a écrit des fichiers de configuration (à améliorer) pour la syntaxe *Therion*. Ils sont disponibles avec la procédure d'installation associée sur <https://github.com/robertxa/Therion-TextWrangler>.

### **Brackets**

[Brackets](#) est un logiciel libre pour MacOS, Windows et Linux

Xavier a écrit des fichiers de configuration (à améliorer) pour la syntaxe *Therion*. Ils sont disponibles avec la procédure d'installation associée sur <https://github.com/robertxa/Therion-Brackets>.



## **KatePart (Kate, KWrite)**

C'est un logiciel libre pour Linux. La coloration syntaxique pour les fichiers *Therion* se trouve sur <https://Therion.speleo.sk/wiki/media/contrib:kateTherion.zip>

## **Emacs**

Pour les habitués à ce logiciel, une coloration Syntaxique *Therion* pour ce logiciel est disponible (avec les instructions d'installation) : <https://Therion.speleo.sk/wiki/media/contrib:Therion-emacs.zip>

## **Vi – Vim**

Encore pour les habitués, il est possible d'utiliser Vi ou Vim pour travailler avec les fichiers *Therion*. La coloration syntaxique peut être installée à partir de <https://Therion.speleo.sk/wiki/media/contrib:Therion-vim.zip>

Plus d'informations se trouve sur la page dédiée du [Wiki Therion](#)

### **• Aides Therion disponibles sur le net**

Sur internet, il est possible de trouver pas mal de tutoriels pour utiliser le logiciel *Therion*. La plupart du temps, c'est en anglais, ce qui n'est pas toujours évident pour le spéléo francophone.

Les deux références d'aide principales sont :

- d'une part le manuel de référence décrivant *Therion* et ses fonctions, le [thbook.pdf](#). C'est à lire et à garder sous la main pour pouvoir le consulter en fonction de ses besoins. Attention, ce n'est pas lui qui va vous apprendre à utiliser *Therion*. Il est aussi fournit lors de l'installation de *Therion*.
- d'autre part le [wiki Therion](#), principalement en anglais, mais en constante évolution et déjà très fourni. Nous ne pouvons que conseiller de le consulter régulièrement !

En français, il est possible de se référer :

- au [wiki Therion francophone](#) ; Il y a pas mal d'informations, mais ce n'est pour l'instant pas complet,
- à une tentative de [traduction du thbook](#).

En anglais, les pages suivantes peuvent aussi être utiles :

- <http://cp.cavesurveying.org.uk/index.php/articles/3-Therion-tutorial> et les liens qui y sont contenus,
- <http://wsc.darkegem.com/footleg/Therion/>

Et aussi, il est possible de se tenir au courant au sujet de *Therion*, et de poser des questions sur [la liste de diffusion e-mails de Therion](#). Elle est en anglais, mais il y a quelques participants francophones qui répondent en français si besoin.

## **4. QU'EST-CE QUE THERION ? QU'EST-CE QUE XTHERION ?**

### **• De quoi a besoin Therion ?**

Basiquement, *Therion* a besoin de différents types de fichiers pour pouvoir produire des topographies. Nous verrons plus tard comment les produire, mais voilà déjà une première description de ces fichiers.

Des templates d'exemples sont disponibles sur [https://github.com/robertxa/Th-Config-Xav/tree/master/Template\\_cave](https://github.com/robertxa/Th-Config-Xav/tree/master/Template_cave). Ils peuvent être utilisés pour faciliter le travail.

### **Les fichiers de données (extension .th)**

Les fichiers de données sont les fichiers qui vont contenir toutes les données brutes de la topographie :

- le nom de la topographie
- la définition des entrées et leurs coordonnées
- Les différentes séances topographiques, avec pour chaque séance :
  - les dates d'exploration et de topographie
  - les auteurs des explorations et des topographies
  - la définition des instruments utilisés pour la topographie, leurs unités et leur calibration si besoin
  - Le type de données topographiques (points, longueurs, compteurs, azimuth pentes, profondeurs, dimensions,...)
  - Le tableau de données de la séance, qui suit le format défini par le type de données

Par défaut, l'extension des fichiers de données est .th.

Dans le cas de cavités complexes et/ou de systèmes karstiques, Certains fichiers .th en contiennent pas de données de séances topos, mais servent à lier les différentes données entre elles. Par exemple, ils permettent d'imposer des jonctions entre deux cavités que nous venons de jonctionner. Nous pouvons donc travailler le dessin de chaque cavité ou de chaque zone explorée de façon indépendante, et de fusionner le tout par la suite grâce à ces fichiers.

Ces fichiers peuvent se construire à partir de rien à l'aide d'un éditeur de texte, ou en adaptant la copie d'un fichier préexistant, ce qui facilite grandement leur construction.

### **Les fichiers de commandes de compilation (extension .thconfig)**

La philosophie de *Therion* n'est pas de dessiner pas une topographie comme sous un logiciel vectoriel classique, en voyant directement le résultat. Au contraire, nous dessinons uniquement des lignes, des points et des aires auxquels nous associons des attributs. Nous utilisons alors une banque de figurés issus de banques homogénéisées officielles (par exemple, la banques UIS) qui correspond au attribut donné lors du dessin. Nous ne pouvons obtenir une carte finale qu'après la compilation du projet qui va interpréter puis transcrire en figurés les points, lignes, et aires en fonction de leurs attributs.

Pour cette compilation, de nombreuses options sont possibles, dépendamment de notre but :

- Quels fichiers de données, de liaisons, et de dessins voulons-nous utiliser pour la production finale ? Quel fichier MNT ? Quelle image de drapage ?
- Par exemple, si nous voulons produire une carte, quelles sont ses caractéristiques ?
  - Quelle vue (plan, coupe développée, coupe projetée) voulons nous ?
  - Quelle projection géographique ? Quel système de référence ?
  - Quelle échelle ?
  - Quelle langue ?
  - Quelles couleurs ?
  - Quelle banque de figurer utiliser ?
  - Veut-on une grille ? Si oui, de quelle taille ?
  - Veut-on un titre ? Une description ? Une légende des figurés utilisés ? Un copyright ?

- Quel type d'export voulons-nous après la compilation ?
  - Une carte finale en plan ? en Coupe développée ? En coupe Projetée ?
  - Un atlas ?
  - Un modèle en 3-D ?
  - Des fichiers intermédiaires pour le dessin ?
  - Un fichier kml qu'on puisse intégrer dans Google Earth ?
  - Des Shapefiles pour utiliser dans un logiciel SIG ?
  - Des listes de synthèses (quelles cavités, quelle spéléométrie, coordonnées des cavités, liste des points d'interrogation,...)
  - La base de données sql du système ?
  - Sous quel(s) format(s) ? (pdf, dxf, txt, html,...)

Ces questions ne sont pas exhaustives, ce ne sont que des exemples. Mais justement, un fichier de compilation est là pour permettre d'obtenir un produit final en adéquation avec tout ce questionnement. Beaucoup d'options ont des valeurs par défaut, un fichier de compilation peut donc être très simple. Les fichiers de compilation possèdent une extension *.thconfig*.

Comme pour les fichiers de données, ces fichiers peuvent se construire à partir de rien à l'aide d'un éditeur de texte, ou en adaptant la copie d'un fichier préexistant, ce qui facilite grandement leur construction.

### **Les fichiers de dessins (extension .th2)**

Les fichiers de dessins peuvent être édités dans un éditeur de texte pour correction, mais pour les produire, il est nettement plus pratique de le faire à l'aide de l'éditeur de dessin dans le logiciel *XTherion*.

Ces fichiers de dessins possèdent une extension *.th2*.

Ils vont contenir :

- le squelette de la topographie que nous voulons dessiner (ce n'est pas obligatoire)
- un dessin au format jpg ou gif, avec soit une échelle, soit la position de quelques stations topographiques
- les différents calques (appelés scraps en langage *Therion*) dessinés
- et les dessins des points, des lignes et des aires, avec leurs attributs respectifs

### **Les fichiers intermédiaires (optionnel)**

Des fichiers intermédiaires peuvent être produits par la compilation (si nous le demandons, évidemment !). C'est notamment le cas pour les fichiers vectoriels du squelette topographique.

Ces fichiers sont au format *.xvi* (texte, mais spécifiquement pour *Therion*). Ils possèdent la ligne de cheminement, les stations topographiques, et les dimensions à chaque station topographique (gauche, droite, haut, bas ou les visées d'habillage).

Leur intérêt est que nous pouvons les importer dans un fichier dessin, et les utiliser comme base pour pouvoir dessiner directement sur l'écran. Ils sont très utiles pour définir dans le dessin les stations topographiques, et ainsi ancrer le dessin sur le squelette topographique.

Ce sont des fichiers texte, mais il n'y a aucun intérêt à chercher à les éditer manuellement avec un éditeur de texte.

### **Les fichiers de configuration (optionnel)**

Enfin, il est possible de construire des fichiers de configuration que nous appellerons dans les fichiers de compilation. Ils peuvent avoir deux intérêts :

- simplifier les fichiers de compilation, et du coup les rendre plus lisible,
- homogénéiser le dessin final de toutes les topographies d'un même système

Ces fichiers fonctionnent comme les fichiers de compilation sans les définitions d'exports. Ils sont au format texte, avec l'extension .thc.

Comme pour les fichiers de données ou de compilation, ces fichiers peuvent se construire à partir de rien à l'aide d'un éditeur de texte, ou en adaptant la copie d'un fichier préexistant, ce qui facilite grandement leur construction.

### • Que peut produire *Therion* ?

Voilà la question... Et une des forces de cette suite logicielle. *Therion* peut produire plein de choses, et pas uniquement des cartes... Ne pas hésiter à visiter le [site internet de \*Therion\*](#) pour avoir une idée de quelques exports possibles.

#### ***Des cartes topographiques***

C'est la première idée qui vient à l'esprit... Et en fait, c'est souvent pour la production de cartes topographiques de qualité que les spéléologues d'exploration vont venir vers *Therion*.

En effet, *Therion* permet la production de cartes topographiques, que ce soit en plan, en coupe développée ou projetée. Ces cartes peuvent être exportées par *Therion* en pdf, svg, xhtml, ou dxf.

Nous y pensons moins, mais il est aussi extrêmement facile d'exporter un atlas (livre contenant la topographie découpée et ordonnée en feuilles A4) au format pdf.

#### ***Des fichiers topographiques 3D***

Nous pouvons exporter aussi les topographies en 3D sous différents formats :

- format .lox pour visualisation avec Loch
- format .3d pour visualisation avec Aven (issu de la suite logicielle libre Survex)
- format .plt pour Compass
- dxf
- ESRI,
- vrmf
- 3dmf
- kml pour Google Earth

Si un MNT a été fourni lors de la compilation, alors, il est possible d'afficher la topographie en trois dimensions avec le relief au-dessus.

#### ***Des fichiers géoréférencés pour logiciels SIG***

La puissance de *Therion* réside aussi dans le fait qu'il peut exporter des fichiers vectoriels géoréférencés en 2D ou en 3D pour leur utilisation dans un logiciel SIG, et pour par exemple permettre des études hydrogéologiques.

Le format le plus commun est le format shapefile ESRI, mais aussi le dxf et le kml.

Ce point est d'autant plus intéressant lorsque nous travaillons non pas sur une grotte en particulier, mais sur tout un système karstique.

Une vraie base de données et des exports prédéfinis issus de cette base de données

Enfin, et c'est là que *Therion* diffère des autres logiciels topographiques, c'est que la base de tous ses exports est une base de données globale et géoréférencée issue des données topographiques.

Bien évidemment il est possible d'exporter cette base de données globale afin de pouvoir l'interroger avec un logiciel de base de données adéquat. Le format de cet export peut soit être du sql, soit du csv.

Un corolaire est qu'il est alors possible d'exporter des listes de statistiques provenant de cette base de données. Dans les listes prédéfinies dans les exports *Therion*, nous pouvons souligner la liste des cavités/entrée/topographies, et la liste des continuations. Leur format peut être du html, du txt, du dbf ou du kml pour affichage dans Google Earth (très utile pour les listes de continuations !).

- **Therion ? XTherion ? Loch ?**

### ***Therion***

*Therion* est le logiciel en commande en ligne qui permet de compiler les topographies.

En effet, nous ne dessinons pas une topographie comme sous un logiciel vectoriel classique, mais juste des lignes, des points et des aires auxquels nous associons des attributs. La compilation permet de produire une carte finale, avec interprétations puis transcription en figurés issus de banques homogénéisées officielles (par exemple, la banques UIS) des points, lignes, et aires en fonction de leurs attributs.

### ***XTherion***

*XTherion* est une interface graphique pour *Therion*.

Il permet :

- l'édition des fichiers textes *Therion* (fichiers de configurations, fichiers de données)
- la production de dessins au format *Therion*
- la compilation pour la production de cartes, listes, fichiers SIG,...

### ***Loch***

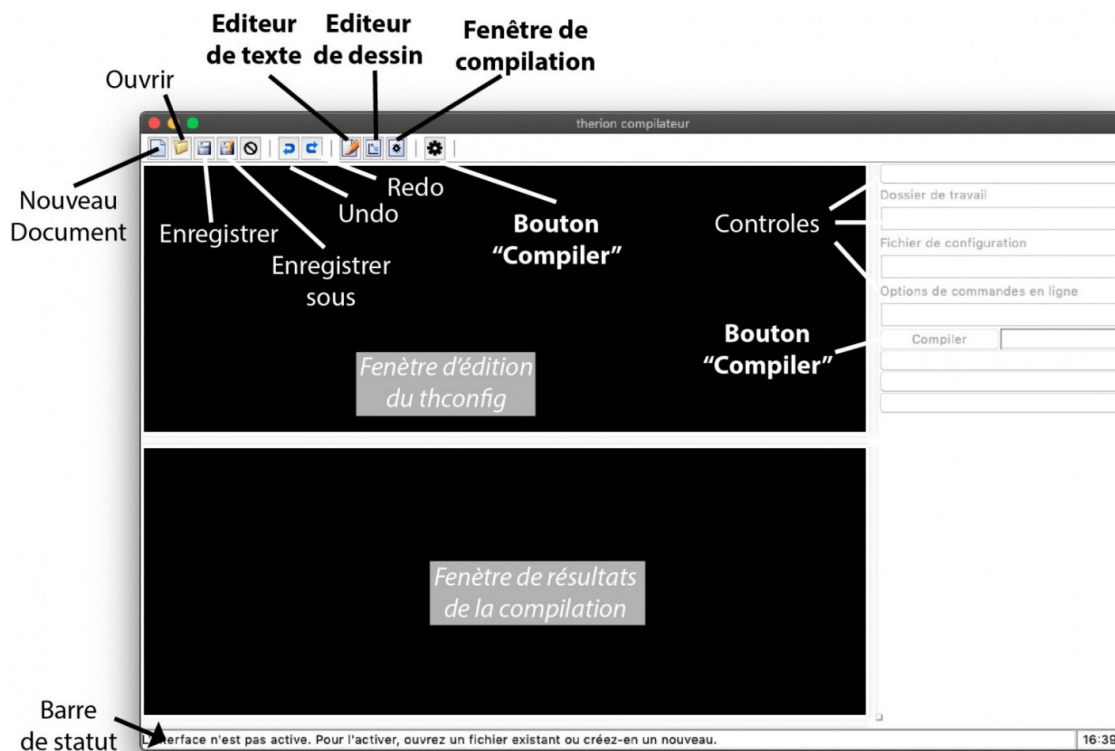
Loch est le visualisateur 3-D des sorties *Therion* en 3D. Il permet de manipuler les fichiers *Therion* 3-D au format .lox

- **Comment utiliser le logiciel XTherion ?**

Nous pouvons utiliser le logiciel *XTherion* pour trois actions distinctes :

- effectuer la compilation à partir d'un *.thconfig* chargé,
- modifier les fichiers *Therion* (.th, *.thconfig*,...) dans l'éditeur de texte,
- dessiner une topographie.

Chacune de ces actions s'effectue dans une fenêtre spécifique. A l'ouverture de *XTherion*, nous obtenons cette fenêtre (ici annotée) :



### La fenêtre de compilation

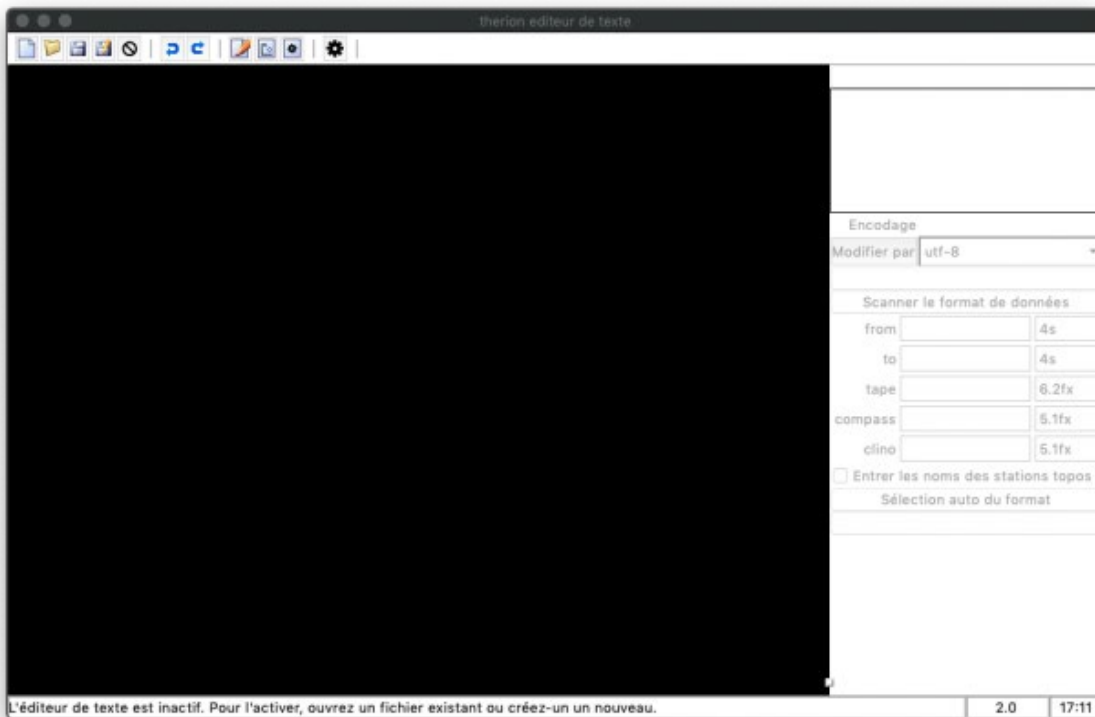
Les trois boutons « **Editeur de texte** », « **Editeur de dessin** » et « **Fenêtre de compilation** » permettent de changer type de fenêtre, dépendamment de ce que nous voulons faire.

Sur toutes les fenêtres, le bouton rond-barré permet de fermer le fichier ouvert dans ladite fenêtre.

La fenêtre de compilation est divisée en trois parties :

- En haut, une fenêtre d'édition. Lorsqu'un fichier de compilation *.thconfig* est chargé, il s'affiche ici. Il est alors possible de le modifier. Si nous voulons écrire un nouveau fichier, nous pouvons le faire après avoir appuyé sur le bouton « nouveau ». Nous pouvons alors écrire dans cette fenêtre.
- En bas, une fenêtre vide, dans laquelle nous ne pouvons rien faire. C'est ici que sera affiché tous les informations de la compilation (ce que fait la compilation, les erreurs s'il y en a, les statistiques,...)
- Sur la droite, un panneau de contrôle. Ici, il y a aussi un bouton « compiler » qui fait la même chose que le bouton en forme de roue dentée dans le bandeau supérieur. Lorsqu'on clique sur l'un des deux, si un fichier de compilation *.thconfig* est chargé, *Therion* effectue la compilation. A l'issue de la compilation, la case à droite change de couleur. Elle devient verte si la compilation c'est bien passée, elle devient orange s'il y a un warning (c'est à dire qu'il y a une erreur, généralement dans le dessin, mais qu'elle ne plante pas la compilation), et elle devient rouge s'il y a une erreur...

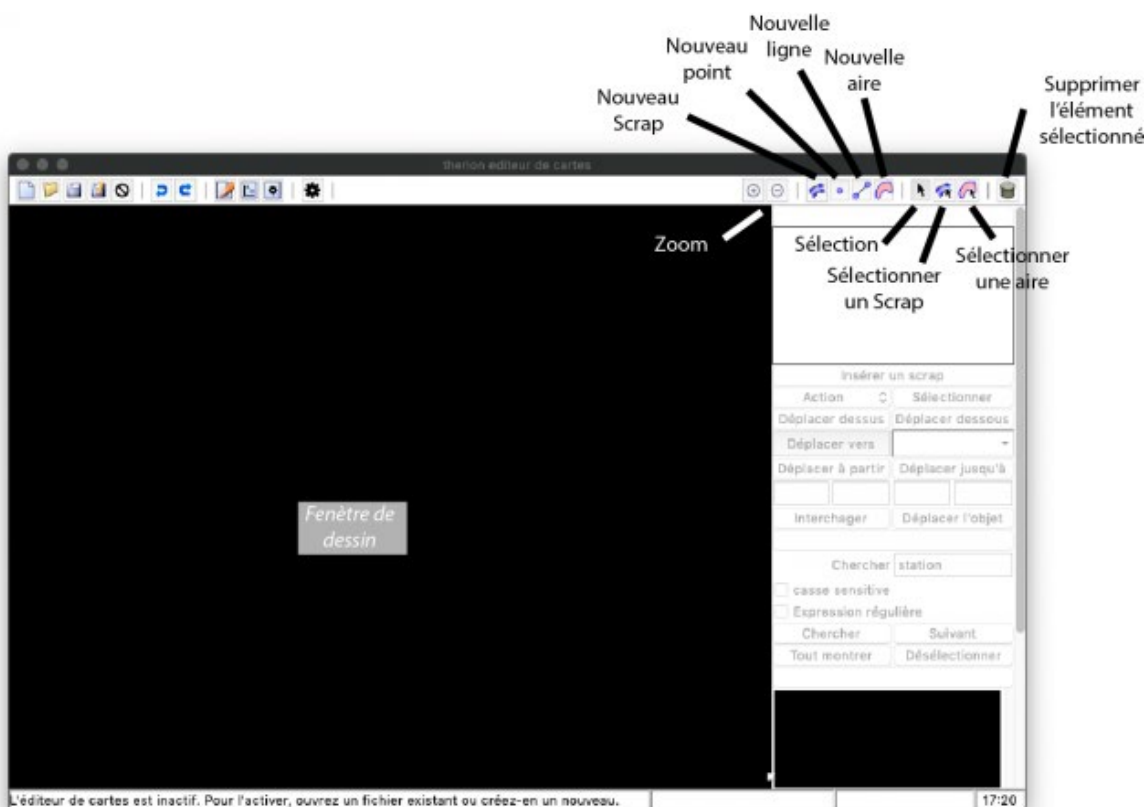
L'éditeur de texte accessible en cliquant sur le bouton adéquat du bandeau se présente quand à lui ainsi :



### L'éditeur de texte intégré à XTherion

Les boutons sont semblables à la fenêtre de compilation. Si besoin, l'action « compiler » est toujours accessible en appuyant sur la roue crantée du bandeau supérieur. A droite, le panneau de contrôle facilite l'entrée de nouvelles données.

Enfin, nous accédons à la fenêtre de dessin en cliquant sur le bouton adéquat du bandeau supérieur. Cette fenêtre est la plus complexe, mais nous la décortiquerons en détail plus tard. Mais son utilisation est très logique, et nous prenons rapidement l'habitude de l'utiliser.



## **L'éditeur de dessin**

La plus grande partie de l'éditeur de dessin est occupée par l'espace où nous allons dessiner.

Les boutons les plus importants sont ceux du bandeau supérieur à gauche. Ils permettent :

- De créer un nouveau calque (=scrap), des nouveaux points, des nouvelles lignes, des nouvelles aires
- De sélectionner manuellement (la flèche sélection) ou par itération les différents objets du dessin
- De supprimer l'objet dessiné.

Le panneau de contrôle contient des onglets :

- de gestion du fichier de dessin (par exemple, ajouter un squelette topographique, ajouter une image de fond, organiser l'ordre des objets,...)
- pour chaque type d'objets (points, ligne, aires) qu'il permet de contrôler (type, sous-type, options,...)

Mis à part un calculateur d'alcoolémie planqué dans l'interface graphique, Il n'y a rien d'autre... Contrairement à la croyance populaire, nous ne pouvons pas dire que ce soit une interface compliquée, loin de là !

### **• Quelques notions de base**

Avant de pouvoir entrer dans le vif du sujet, il est nécessaire de rappeler quelques notions de base. Elles ne sont pas difficiles (en tout cas, pas plus que d'aligner 3 ou 4 mots cohérents !), mais elles sont indispensables à comprendre et à intégrer.

Comme nous l'avons décrit plus tôt, tous les fichiers d'entrées sont des fichiers textes. Nous pouvons donc les créer, les lire et les modifier avec n'importe quel éditeur de texte. Voilà quelques règles de base qui vous évitera de nombreux problèmes :

#### ***De l'utilisation des noms (de fichiers, de topographies,...)***

Les noms des fichiers textes ne doivent comporter ni d'espace, ni de caractères accentués. Idéalement, leurs noms sont représentatifs de ce qu'ils sont.

Idem, ne pas utiliser d'espace ou de caractères accentués dans les noms des identifiants que nous utiliserons tout au long de la topographie.

#### ***Le séparateur de décimales***

C'est un problème. Dans les langues latines, comme le français, le séparateur de décimales est la virgule. Si vous avez acheté votre ordinateur en France et que vous n'avez pas modifié les paramètres régionaux et la définition du séparateur de décimales, alors, certains (rares) programmes vont considérer que la décimale est bien marquée par une virgule. C'est le cas, par exemple, du logiciel Microsoft Excel, qui est souvent utilisé pour entrer des données topographiques.

Mais voilà, dans les langues anglo-saxonnes, le séparateur de décimales, c'est le point et non la virgule. Et par défaut, la plupart des logiciels (sauf de rares exceptions comme Microsoft Excel) travaillent en convention anglo-saxonne, c'est à dire avec le point. C'est le cas de *Therion*, mais aussi de la plupart des logiciels SIG.

En conséquences, si vous faites un copier-coller de votre tableau de données Excel avec une virgule comme séparateur de décimales, dans un fichier pour *Therion*, et bien vous allez recevoir une belle volée d'insultes.



Alors, faites attention à comment vous séparez vos décimales...

Un moyen très simple de s'affranchir de ce problème, c'est de modifier les préférences de votre système, et d'imposer le point comme séparateur de décimales. C'est simple à faire, et ça évite beaucoup d'erreurs... et de prises de tête...

### **Des mots clefs de commandes**

Dans ces fichiers textes, chaque ligne va correspondre à une commande donnée par un mot clef correspondant à cette commande. Chaque ligne sera donc exécutée lors de la compilation.

Nous devons écrire une commande par ligne. Ne pas écrire plusieurs commandes sur la même ligne.

Quelque soit le fichier, chaque information que nous voulons donner est écrite sous la forme : « commande » <attributs> [-options <attributs de l'option>]. Par exemple, si nous voulons définir « Monsieur Vulcain » comme auteur d'une topographie, la « commande » est le mot « team » et l'attribut est « Monsieur Vulcain ». Nous écrivons dans le bloc d'une séance topographique :

```
team "Monsieur Vulcain"
```

Nous ne pouvons pas écrire juste « team » sans l'attribut parce que les attributs sont obligatoires.

En revanche, les options ne sont pas obligatoires. Si ce Monsieur Vulcain a été la personne à la prise de note et que nous voulons l'indiquer dans les données, nous écrivons :

```
team "Monsieur Vulcain" notes
```

« notes » étant le mot clef spécifique de cette information.

Ceci n'est qu'un exemple, et permet d'illustrer comment nous allons écrire les informations dans les fichiers d'entrées de *Therion*. Toutes les commandes fonctionnent de manière identique. Seul le nombre d'attributs et le nombre d'options change.

Il faut noter que toutes les commandes sont des mots clefs anglais. Mais généralement, ce sont des mots que vous connaissez déjà parce que vous les avez déjà vus par ailleurs. Dans la suite, pour chaque mot clef ou commande utilisé, nous essayerons d'en donner la traduction française pour faciliter l'apprentissage.

### **Comment rajouter un commentaire ?**

Si nous souhaitons ajouter un commentaire (par exemple pour expliquer ce que fait telle ou telle ligne, ou pour qu'une ligne ne soit pas exécutée temporairement), nous pouvons commenter cette ligne. Il suffit alors d'ajouter le caractère dièse « # » au début de la ligne.

Par exemple :

Cette ligne n'est pas un commentaire et va générer une erreur car elle n'est pas une commande non plus

```
# Cette ligne, en revanche, est bien un commentaire, parce qu'elle est précédée d'un "#"
```

```
# Pour continuer un commentaire sur la ligne suivante, il faut de nouveau commencer la ligne avec le signe "#"
```

### **La notion de blocs**

Nous allons voir que la structure de tous ces fichiers est basée sur des définitions de blocs :

- des blocs de topographies (fichiers de données),
- des blocs de séances topographiques (fichiers de données),

- des blocs de mise en page (fichiers de compilation et de configuration),
- des blocs de calques (fichiers de dessin),
- des blocs de lignes (fichiers de dessin),
- des blocs d'aires (fichiers de dessin),
- des blocs de maps (fichiers de données ou de liaisons de données).

Il nous faut apprendre à définir des blocs. Définir un bloc, ce n'est pas difficile, il faut lui dire où il commence, et où il finit. Chaque type de blocs est identifié par un nom spécifique. Pour les fichiers de données, nous utiliserons :

- le bloc topographie, qui se nomme le bloc survey [=topographie] (fichiers de données),
- le bloc séance topographique, qui se nomme le bloc centerline [=ligne centrale/principale] (fichiers de données),
- le bloc de maps qui se nomme map [=carte] (fichiers de données ou de liaisons de données).

Pour le dessin, il y a aussi les blocs :

- de scraps : scrap
- de ligne : line
- d'aires : area

Dans les fichiers de configuration et de compilation, nous trouverons le bloc de mise en page (layout en anglais): layout.

Un bloc sera donc défini au début par le mot clef le définissant, et à la fin par le mot « end » précédant le mot clef (sans espace). Cela donnera par exemple :

```
survey
  Commandes du bloc survey
  centerline
    Commandes du bloc centerline (une séance topographique)
  endcenterline
  centerline
    Commandes du bloc centerline (une autre séance topographique)
  endcenterline
endsurvey
```

Ce n'est pas obligatoire, mais une bonne chose est de marquer avec une indentation (quelques espaces ou une tabulation) ce qui est à l'intérieur d'un bloc. A chaque bloc imbriqué, nous augmentons alors d'une indentation.

## 5. ENTRER DES DONNÉES TOPO POUR THERION

### • Comment créer un fichier de données ?

Cette page décrit la structure d'un fichier de données topographiques.

#### **Structure d'un fichier de données topographiques**

Dans le cadre d'un fichier de données topographiques, nous allons utiliser les blocs de topographie et de séance topographique. Cela donnera par exemple :

```
survey <identifiant>
  Commandes du bloc survey
  centerline
    Commandes du bloc centerline (une séance topographique)
  endcenterline
```

```
centerline
  Commandes du bloc centerline (une autre séance topographique)
endcenterline
endsurvey
```

Dans cet exemple, il y a deux points importants :

1. le mot clef survey est suivi d'un nom. C'est l'identifiant de la topographie. Il est impérativement sans espaces. Il est important/fondamental parce que par la suite, nous nous référerons à cette topographie en l'appelant par son identifiant. C'est ainsi que nous allons pouvoir savoir avec quelle topographie le dessin que nous sommes en train de dessiner va s'ancrer.
2. nous avons imbriqué un bloc centerline à l'intérieur du bloc survey. En fait, les blocs centerline ne peuvent pas être en dehors des blocs survey, ils sont forcément à l'intérieur. En conséquence, un bloc centerline ne peut pas dépendre de deux (ni plus) blocs survey différents. En revanche, un bloc survey peut contenir autant de bloc centerline que nécessaire !

Ce n'est pas obligatoire, mais une bonne chose est de marquer avec une indentation (quelques espaces ou une tabulation) ce qui est à l'intérieur d'un bloc. A chaque bloc imbriqué, nous augmentons alors d'une indentation.

### **Structure d'une séance topographique (centerline)**

Chaque séance topographique (ou chaque série topographique si nous travaillons en séries) est définie par un bloc centerline. Nous le rappelons, ce bloc centerline doit obligatoirement être inclus dans un bloc survey.

Ce bloc centerline va être divisé en deux parties :

- une première partie avec toutes les informations sur comment a été faite la topographie
- une seconde partie qui correspond au tableau de données en lui même

Nous allons décortiquer chaque partie avant de donner un exemple simple.

```
centerline -title "Exemple"
```

C'est le début de la séance topographique d'exemple. Nous lui donnons un titre (facultatif) « Exemple »

### **Définition de comment a été levée la topographie**

Il est obligatoire de définir ici les instruments que nous utilisons, leurs unités et leur calibration :

- dans le cas d'une topographie effectuée avec un disto, en mètres, et une mesure de la pente en degrés (lenght, c'est pour définir la longueur, compass, l'azimut et clino, la pente):

```
units lenght meters
units compass degrees
units clino degrees
```

- dans le cas d'une topographie effectuée avec un disto, en mètres, et une mesure de la pente en grades :

```
units lenght meters
units compass grad
units clino grad
```

- dans le cas d'une topographie effectuée avec une boîte topo-Vulcain en grades :

```
units length meters
units compass grad
units clino grad
calibrate counter 0 1.00
calibrate clino 100
units counter centimeters
```

La commande « calibrate » s'utilise ainsi :

```
calibrate <quantity list> <zero error> [<scale>]
```

où :

- <quantity list> = qu'est-ce qu'on veut calibrer. Cela peut prendre les valeurs length (= longueur), tape (= ruban), bearing (= position), compass (= azimuth), gradient, clino, counter (= compteur), depth (= profondeur), x, y, z, position, easting, dx, northing, dy, altitude, dz
- <zero error> = quelle correction nous voulons apporter. Elle sera calculée ainsi : nouvelle valeur = valeur donnée – zero\_error.
- [<scale>] = quel facteur d'échelle voulons nous appliquer ? dans l'exemple ci-dessous, il est mis à 1 pour le compteur, ce qui veut dire que 1 tour de compteur est équivalent à 1 unité, soit 1 cm. Avec la dernière version de la boîte topo Vulcain, ce facteur est de l'ordre de 0.5.

Accessoirement, nous pouvons aussi définir

- la date de la topographie (année.mois.jour) ; elle est utilisée pour calculer automatiquement la déclinaison magnétique ; si nous ne la fournissons pas, il faut définir la déclinaison

```
date 2020.04.12
```

- la déclinaison magnétique (dans le cas où la date de la topographie n'est pas donnée) ; ne pas oublier l'unité :

```
declinaison 0.5 degrees
```

- l'équipe qui a fait la topographie (autant de lignes que de personnes ; notez l'utilisation du « / » pour délimiter prénom / nom)

```
team "Monsieur Vulcain"
team "Monsieur/de Vulcain"
```

- la date de l'exploration

```
explo-date 2020.04.12
```

- l'équipe qui a fait l'exploration

```
explo-team "Monsieur Vulcain"
explo-team "Monsieur/de Vulcain"
```

- la station qui correspond à(aux) l'entrée(s) (ici 0.0 ; une ligne par entrée)

```
station 0.0 "Entrée principale" entrance
```

- le système de coordonnées qu'on va utiliser pour définir les coordonnées de l'entrée (cs = coordinates system, soit système de coordonnées). L'argument de la commande cs peut être
  - le système de coordonnées géographiques lat-long ou long-lat ; en ce cas, le système de référence est WGS84.
  - quasiment tous les systèmes de projections existant, que nous pouvons appeler
    - par leur nom (ex : UTM32N)
    - par leur code EPSG (EPSG:<numéro\_EPSG>)
    - par leur code ESRI (ESRI:<numéro\_ESRI>)
    - et quelques autres très spécifiques

cs lat-long

- les coordonnées de l'entrée (fix station lat long alt [erreur\_lat erreur\_long erreur\_alt])

fix 0.0 4.2345 46.9876 1200

### **Déclaration du tableau de données topographiques**

Avant d'insérer le tableau de données topographique, il faut définir la structure du tableau de données, c'est à dire à quoi correspondent les différentes colonnes. Cela se fait grâce à la commande data suivie du style de données puis de ce que représente chaque colonne, dans l'ordre que nous allons utiliser. Voici quelques exemples de définition de tableaux de données :

Nous voulons entrer un tableau de données normales/classiques qui contient :

- la station de départ (mot clef = `from`)
- le station d'arrivée (mot clef = `to`)
- la longueur entre les deux stations (mot clef = `length`)
- l'azimut (en direct ; mot clef `compass`),
- la pente (en direct ; mot clef `clino`),
- puis les dimensions gauche droite haut bas (respectivement `left`, `right`, `up`, `down`)

La commande sera :

```
data normal from to length compass clino left right up down
```

Si finalement, l'azimut et le clino ne sont pas en directs, mais en inverses, il suffit de modifier la ligne en :

```
data normal from to length backcompass backclino left right up down
```

Noter le « back » (= arrière) ajouté avant compass et clino pour indiquer que leur mesure est en inverse.

Si nous voulons entrer des données issues d'une boîte topo Vulcain, nous pouvons déclarer le tableau de données ainsi :

```
data topofil from to fromcount tocount compass clino left right up down
```

où « fromcount » et « tocount » correspondent respectivement au compteur au point de départ et au compteur au point d'arrivée.

Lors de l'utilisation d'une boîte topo Vulcain, le clino se lit de 0 à 180° (ou 0-200 grades). En ce cas, la déclaration de tableau ci-dessus renverra une erreur sur le clino, comme quoi les valeurs

ne sont pas comprises entre  $-90^\circ$  et  $+90^\circ$  (ou  $-100/+100$  en grades). Une façon de corriger cela est, avant cette ligne « data », d'utiliser la commande calibrate ainsi :

```
calibrate clino -90 # si unités du clino en degrés  
calibrate clino -100 # si unités du clino en grades
```

De manière générale, la commande « data » s'écrit :

```
data <style> <readings order>
```

Ici, <style> peut prendre un des arguments suivant (c'est ici de la syntaxe Survex) :

- normal : c'est ce que nous utilisons lors d'une topographie avec un DistoX
- topofil : cette fois, c'est ce que nous utilisons lors d'une topographie avec une boîte Topo Vulcain
- diving : pour la plongée
- cartesian : si nous donnons des coordonnées de points de type (x,y,z), dans un référentiel cartésien
- cypolar : La même chose, mais dans un référentiel polaire
- dimensions : si nous voulons donner uniquement des dimensions à un point donné
- nosurvey : s'il n'y a pas de données topographique

<readings orders> va décrire les différentes colonnes du tableau avec potentiellement ces termes :

- station
- from
- to
- tape/length
- [back]compass/[back]bearing
- [back]clino/[back]gradient
- depth
- fromdepth
- todepth
- depthchange
- counter
- fromcount
- tocount
- northing
- easting
- altitude
- up/ceiling
- down/floor
- left
- right
- ignore
- ignoreall

### ***Le tableau de données topographiques***

Ensuite, il nous faut entrer le tableau données. C'est du format texte, ce qui fait que pour matérialiser les colonnes, il y a plusieurs possibilités très simples. La séparation des colonnes peut-être :

- un simple espace
- une tabulation

L'avantage d'utiliser une tabulation, c'est que ça permet de visualiser le tableau sous forme de colonnes. C'est la solution que nous préconisons, mais ce n'est pas obligatoire de la suivre.

Par exemple, dans le cas de données issue d'une topographie effectuée avec un disto (X), chaque ligne de données ressemblera à quelque chose comme :

```
2018-JR-22 2018-JR-23 2.5 39 -42 0.6 1.5 8 1.2 # Base du P12
```

où

- 2018-JR-22 est la station de départ,
- 2018-JR-23 est la station d'arrivée
- 2.5 est la distance entre les deux stations
- 39, l'azimut
- -42, la pente
- et 0.6 1.5 8 1.2 les dimensions gauche droite haut bas
- # Base du P12 est un commentaire facultatif lié à la station d'arrivée

Pour les habitués, dans Visual Topo, il faut systématiquement commencer chaque tableau de données par une ligne « vide » 0 0 0 0. Avec *Therion*, ça ne fonctionne pas ainsi. Il ne peut pas y avoir de visé d'une station à cette même station. Par exemple :

```
0 0 2.5 39 -42 0.6 1.5 8 1.2
```

va générer une erreur lors de la compilation.

En revanche, si nous modifions une des stations 0 en 0.0, 0.1, junk,... tout se passe très bien :

```
0 0.1 2.5 39 -42 0.6 1.5 8 1.2 # OK
# ou
0 junk 2.5 39 -42 0.6 1.5 8 1.2 # OK
```

Si l'une des stations est nommée par « - » ou « . », alors, cette ligne va être interprétée comme visée d'habillage, et non visée de cheminement topographique.

### **Comment entrer les données ?**

Entrer les données peut-être très fastidieux. Evidemment, il faut les entrer dans le même ordre que ce qui a été défini par la ligne « data ».

Si vous avez l'habitude d'écrire et d'utiliser la tabulation, alors, vous pouvez les entrer directement à la main dans l'éditeur de texte. Ainsi, nous devons les entrer ligne à ligne.

Si au contraire, vous êtes habitués à les entrer colonne par colonne, alors, utiliser un tableur de type Excel qui facilite grandement cette façon d'entrer les données. Ensuite, faites un copier-coller dans le fichier de données *Therion*. Automatiquement, la séparation des colonnes sera une tabulation. Attention, dans ce cas, prenez garde aux séparateurs de décimales....

Vous pouvez aussi noter que pour que vos données soient lisibles (par exemple si vous voulez visuellement isoler quelques visées correspondant à la topographie d'un affluent), vous pouvez sauter des lignes et les laisser blanches, cela ne pose aucun problème.

`endcenterline`

Voilà, c'est la fin d'un bloc de séance topographique d'un fichier de données.

Dans le cas d'une topographie effectuée « à l'ancienne » à la main, il faut se taper tout ce processus. Pour cela, il peut être bon de faire un copier-coller d'un précédent fichier de données, et de l'adapter au nouveau jeu de données, ce qui n'oblige pas à retenir tous les mots clefs, ni à les réécrire à chaque fois ! C'est même très rapide !

Dans le cas d'une prise de note électronique (comme avec TopoDroid par exemple), ce fichier est automatiquement construit. Il suffit de l'éditer pour modifier les options non essentielles que nous voulons définir/modifier.

Un fichier d'exemple

Voici un fichier de données exemple d'une topographie contenant deux séances topographiques.

### • **Comment ajouter une nouvelle séance topographique ?**

Et voilà, nous revenons d'une cavité où nous avons trouvé (et topographié !) une suite. La question est de savoir comment nous pouvons l'intégrer à la topographie précédente.

Il y a deux solutions possibles :

- Nous éditons le précédent fichier de données topographiques, et à l'intérieur du même bloc de topographie (« survey »), nous faisons un nouveau bloc de séance topographique (centerline). Ceci fonctionne bien pour les petites cavités
- Si la cavité est importante, alors, il peut être intéressant de travailler le nouveau bout topographié de façon complètement indépendante

#### ***Rajout de la nouvelle séance topographique dans un fichier de données existant***

Ceci est assez simple, il suffit de créer un nouveau bloc « centerline – endcenterline » dans le bloc survey – endsurvey existant en suivant ce qui a été expliqué précédemment.

Ce qui est important, c'est :

- aucune station topographique de cette nouvelle séance topographique ne doit s'appeler comme dans une séance topographique antérieure et dans le même bloc survey – endsurvey,
- sauf la (ou les s'il y a des bouclages ou différentes branches se recordant à une séance précédente) station(s) d'accroche (s) à la topographie précédente.

#### ***Travailler le nouveau bout topographié de façon indépendante***

Dans ce cas, nous créons un nouveau fichier de données indépendant. Dans ce fichier :

- nous définissons un nouveau bloc survey – endsurvey. Nous lui donnons un identifiant différent de l'identifiant que nous avons utilisé précédemment, afin de ne pas se mélanger les pinces (et de ne pas générer d'erreurs à la compilation)
- nous construisons un nouveau bloc de séance topographique centerline – endcenterline de la même manière que pour le fichier de données précédent.

L'avantage de faire ainsi, c'est que nous allons pouvoir travailler le dessin de façon indépendante, et ainsi diminuer le poids des fichiers, et donc le temps de travail.

Le problème, c'est que si nous avons deux bouts de topos indépendants mais appartenant à la même cavité, il faut les joindre entre eux pour pouvoir obtenir la topographie de la cavité complète. Ceci s'effectue finalement assez simplement.

Supposons que nous ayons deux fichiers de données topographiques, que nous appellerons bout-trou1.th (qui contient la survey avec l'identifiant galerie1) et bout-trou2.th (avec la survey galerie2). Supposons aussi que ces deux bouts de topographie jonctionnent en deux stations spécifiques comme, par exemple :

- la station 3 de la survey galerie1 correspond à la station 3 de la survey galerie2
- la station 10 de la survey galerie1 correspond à la station argile27 de la survey galerie2



Pour les joindre, le plus simple est de créer un nouveau fichier topoglobale.th, que nous allons appeler fichier de jonctions maître.

Le contenu de ce fichier de jonction comportera pour le moment deux parties :

- une partie où nous allons déclarer les différents fichiers de données que nous voulons utiliser ; ceci se fait grâce à la fonction « input <[chemin/du/dossier/]nom\_de\_fichier>« .
- une autre partie où nous allons définir les points de jonction ; dans notre cas, cela se fait grâce à la commande « equate 3@galerie1 3@galerie2 » ; Si nous voulons que *Therion* exporte les statistiques pour les deux fichiers de données sans les séparer, alors, nous devons entrer cette (ces) commande(s) equate dans un bloc survey – endsurvey.

Le fichier topoglobale.th contiendra donc :

```
# déclare la nouvelle survey qui va être la somme des deux surveys
galerie1 et galerie2 ; # nous l'appelons MaGrotte
survey MaGrotte

# Appel des fichiers de données
input bout-trou1.th
input bout-trou2.th

# impose la première jonction
equate 3@galerie1 3@galerie2
# impose la seconde jonction
equate 10@galerie1 argile27@galerie2

#fin de la survey
endsurvey
```

Il est à noter que pour la compilation, dans le fichier de compilation, nous appellerons ce fichier maître topoglobale.th et non pas chaque fichier de données topographique parce que ce fichier maître appelle lui même les fichiers topographiques.

Dans le cas de cavités complexes, voir de systèmes karstiques, il est ainsi facile de mettre sur la même topographie plusieurs bouts de topographie, voir même plusieurs cavités. Ce fichier maître peut lui-même être appelé par fichier maître supérieur pour monter encore d'un niveau dans la production des produits finaux.

La seule contrainte, importante, est qu'il faut que *Therion* puisse localiser chaque entité chargée par ces fichiers de jonctions. Cela peut être fait de deux manières :

- Comme dans l'exemple que nous venons de donner, nous jonctionnons deux bouts de cavités grâce aux commandes equate.
- Si nous voulons mettre ensemble deux grottes qui ne sont pas jonctionnées, alors, ceci est possible si et seulement si chaque des grottes possède dans une de ses instances centerline – endcenterline une déclaration de système de coordonnées (commande « cs ») associée à la déclaration des coordonnées d'une ou plusieurs entrées (commande « fix <station> <coordonnées>« ).

Nous reviendrons plus tard sur ce fichier là, parce qu'il peut aussi servir à l'appel des différents fichiers de dessins, et à définir les jonctions des calques/scraps afin d'obtenir un rendu final de qualité.

## 6. CRÉER UN FICHIER DE CONFIGURATION POUR LA COMPILATION

## • Qu'est-ce qu'un fichier *.thconfig* ?

Un fichier *.thconfig* est le fichier qui donne toutes les instructions pour produire un document final. Il est composé de 3 parties :

1. une première partie qui indique les fichiers de données et de dessin qu'il va falloir utiliser pour générer un produit final
2. une seconde partie qui définit les propriétés de la sortie
3. une dernière partie qui indique quels exports nous voulons.

### **Appel aux fichiers de données et de dessin à utiliser**

Au début du fichier *.thconfig*, nous appelons les fichiers grâce à la commande source :

```
source bout-trou1.th # le fichier de données topo
source bout-trou1.th2 # le fichier de dessin

# éventuellement un fichier de relief (MNT)
source mnt.th
```

Evidemment, c'est dans le cas où ce fichier *.thconfig* est dans le dossier où se trouvent les fichiers appelés. Sinon, il faut indiquer les chemins des fichiers, sans quoi ils ne seront pas trouvés. Par exemple :

```
source MesDonneesTopo/bout-trou1.th
```

Dans le cas d'une cavité complexe ou d'un système de cavité, nous pouvons remplacer l'appel aux données topos et aux dessins par un fichier maître comme le précédent *topoglobale.th* qui contiendra les appels à fichiers (topos et dessins), ainsi que les jonctions de topos et de dessins.

Le début du fichier *thconfig* deviendra alors :

```
source topoglobale.th

# éventuellement un fichier de relief (MNT)
source mnt.th
```

### **Mise en page et propriétés de la sortie**

La partie mise en page est contenu dans un bloc de mise en page, le bloc

```
layout <identifiant>
  # Instructions du bloc layout
endlayout
```

L'identifiant (= le nom de la mise en page) est obligatoire parce qu'il sera appelé dans la partie « exports ».

Il est bien évidemment possible définir différents blocs de mises en page, pour chaque export spécifique que nous voulons.

Beaucoup de commandes peuvent être utilisées dans un bloc de mise en page. Elles ne sont pas obligatoire parce qu'elles possèdent toutes des valeurs par défaut qui seront utilisées si elles ne sont pas définies par l'utilisateur.

Les commandes à l'intérieur d'un bloc de mise en page vont varier suivant le type d'export que nous voulons. Nous pouvons les classer en quatre grandes catégories :

- les commandes de gestion du dessin
- les commandes de gestion du cartouche
- les commandes de gestion des pdf
- les commandes de construction d'un atlas topographique

## **La gestion du dessin**

Les commandes de gestion du dessin permettent :

- de définir la langue à utiliser
  - `language fr #` pour le français
- de définir projection, échelles...
  - `# "cs"` Pour projeter la topo dans le système UTM32 nord  
`cs UTM32N`
  - `# "base-scale"` spécifie l'échelle à laquelle nous avons dessiné nos croquis. Par défaut, c est 1/200. Ici, c'est au 1/1000  
`base-scale 1 1000`
  - `# "scale"` : spécification de l'échelle d'impression. La combinaison entre scale et base-scale contrôle la grosseur des lignes, la taille du texte et des symboles... Il peut être intéressant de jouer avec  
`scale 1 1000`
  - `# "scale-bar"` Echelle graphique, ici de 100 m ampleur  
`scale-bar 100 m`
  - `#` Pour faire une rotation  
`rotate 2.25`
- de définir si nous voulons une grille et comment
  - `# "grid"` Spécifier qu'il faut imprimer une grille au dessous de la topo. "Grid" peut prendre les options off (par de grille), bottom (placée en dessous) ou top (placée au dessus)  
`grid bottom`
  - `#` Spécifier le pas de la grille, ici 100x100x100 m (Trois dimensions, oui, ça sert pour la coupe aussi)  
`grid-size 100 100 100 m`
  - `#` Affiche les coordonnées de la grille <off (pas de coordonnées) / all (sur tous les croisements) / border (uniquement au niveau des croix aux bords de la carte)  
`grid-coords all`
- de définir quelle banque de figurés nous voulons utiliser
  - `# "symbol-set"` : charge une banque de figurés préformatés. Les banques existantes sont UIS (Europe), ASF (Australie), AUT (Autriche), BCRA (British), NSS (USA), NZSS (Nouvelle Zélande), SBE (Brésilien) et SKBB. Il est possible d'en créer soi-même si besoin :  
`symbol-set UIS`
  - `#` pour modifier un seul type de symbole :

```
#symbol-assign <point/lien/area/group/special> <symbol-  
type (le nom du symbole)> <symbol-set>
```

- o # Si nous voulons cacher (hide en anglais) un objet particulier, par exemple ici la ligne du cheminement topo :  
symbol-hide line survey

- de définir les couleurs de la topographie

- o # Une couleur de fond, 85% blanc = 15% noir  
color map-bg 85
- o # Une couleur de topo (RVB)  
color map-fg [100 100 80]
- o # Si nous voulons colorer la cavité en fonction de l'altitude, nous pouvons le faire avec la commande (attention, il faut plusieurs calques/scraps à des altitudes différentes)  
color map-fg altitude
- o # la topographie est transparente (on peut voir les galeries sous-jacentes) (Par défaut, donc, pas vraiment besoin de le spécifier)  
transparency on
- o # Définition du pourcentage de transparence, seulement si transparency est "on"  
opacity 75
- o # Si dans les données appelées, nous avons appelé un MNT avec une image drapée dessus, alors, il est possible de choisir si nous voulons l'afficher avec la commande : "surface <top (au dessus)/ bottom (en dessous) / off (pas d'image)>"  
surface bottom
- o # Dans le cas où nous l'affichons, nous pouvons jouer sur son opacité, entre [0-100] ; Par défaut, c'est 70  
surface-opacity 70

### **La gestion du cartouche**

Le cartouche est le cadre associé à la topographie, qui donne des informations sur cette topographie, comme l'échelle, la flèche du nord, les auteurs, la légende...

De base, nous pouvons la gérer avec ces instructions :

```
# "map-header" : Spécifier la position du cartouche. Nous indiquons les  
cordones du point de la topo ou l'on veut le cartouche:  
# 0 0, c'est en bas, a gauche  
# 100 100, c'est en haut, a droite  
# La manchette a des "points cardinaux" : n (en haut-milieu), s (en bas-  
milieu), ne (coin en haut à droite), sw (coin en bas à gauche), etc.  
# Il faut spécifier un de ces points. C'est lui qui sera placé aux  
coordonnées spécifiée.  
# Par exemple, ici, nous pouvons spécifier que le coin en bas à gauche  
(SW) du cartouche sera placés en haut (y = 105) et à gauche (x = 0) de  
la topo :  
map-header 0 105 nw
```

```

# Voulons nous voir (off) ou pas (on ; en ce cas, il sera sur fond
blanc) l'arrière-plan du cartouche ?
map-header-bg off

# Afficher la légende des symboles
# "legend off" = pas de légende ; "legend on" = légende des symboles
utilisés ; "legend all" = tous les symboles de la banque de symboles +
les symboles définis par l'utilisateur
legend on

# Par défaut, la légende est de 14 cm de largeur
legend-width 21 cm

# Par défaut, la légende est affichée sur 2 colonnes. Nous pouvons le
changer pour 3 colonnes, par exemple :
legend-columns 3

# Ajouter un commentaire sous le titre
map-comment "Samoëns, 74, France"

# Afficher les statistiques d'explo par équipe/nom. C'est lourd
# si la cavité est importante et qu'il y a beaucoup d'explorateurs
statistics explo-length off

# Afficher le développement topographié
statistics topo-length off

# Afficher les copyrights inclus dans les dessins
statistics copyright all

```

Avec un usage avancé, il est possible de modifier en profondeur ce cartouche.

### ***La gestion des données des pdf***

Il est possible d'inscrire des métadatas dans les pdf (qui apparaîtront dans les propriétés du document). Par exemple :

```

# Définition du titre
doc-title "Ma cavité"

# Définition de l'Auteur
doc-author "Auteur du document"

# Définitions de mots clefs
doc-keywords "mots clef 1, mots clefs 2..."

# définition du sujet
doc-subject "description du sujet"

# Autoriser (on) ou non (off) les calques dans le pdf :
layers on

```

Les commandes du bloc layout sont nombreuses. Nous n'avons développé ici que celles qui sont utilisées le plus souvent. Pour en savoir plus, il faut se référer au thbook.

Nous pouvons aussi écrire plusieurs blocs de mise en page dans le même fichier de compilation. Généralement, nous en avons au moins trois :

- un premier, très court et très basique, pour définir l'export du squelette topographique (xvi) qui nous sera utile pour le dessin,

- un second pour le plan pdf
- un troisième pour la coupe pdf
- et parfois un quatrième pour l'atlas

### Quels exports ?

Comme décrit précédemment, nous pouvons exporter différents types de produits :

- une carte en 2 dimensions (map)
- un modèle en 3 dimensions (model)
- un atlas en 2-D (atlas)
- des listes (cave-list ; survey-list ; continuation-list)
- la base de données sql (database)

Ces exports se font au moyen de la commande :

```
export <type> <-options>
```

Le type correspond à map, model, atlas, cave-list, survey-list, continuation-list ou database

Les options principalement utilisées sont :

- -output (ou -o) <fichier> : nom du fichier exporté. Par défaut, c'est « cave »
- -format (ou -fmt) <format> : format du fichier exporté. Pdf, dxf, svg, xhtml, Survex, esri, kml, xvi and bbox pour map ; loch, compass, Survex, dxf, esri, vrml 3dmf et kml pour model ; html, txt, kml ou dbf pour les listes ; sql ou csv pour database
- -projection <id> où <id> définit si la projection est plan, extended (coupe développée) ou projected (coupe projetée) ; valable uniquement pour le type map
- -layout <id> : Indique quel bloc de mise en page utiliser (<id> = nom du bloc que nous avons défini) ; C'est pour les types map et atlas

Donc, si nous voulons exporter en pdf le plan d'une grotte, suivant la mise en page que nous appelé « mamiseenpage » et que nous voulons que cet export s'appelle LaGrotte, alors, nous écrirons à la fin du fichier de configuration :

```
export map -proj plan -fmt pdf -layout mamiseenpage -o LaGrotte.pdf
```

Si nous voulons exporter uniquement le squelette xvi de la topographie en coupe développée pour pouvoir dessiner ensuite, nous écrirons :

```
export map -proj extended -fmt xvi -layout mamiseenpageXVI -o LaGrotte.xvi
```

Si nous voulons un export 3D au format lox pour loch, nous rajouterons :

```
export model -o LaGrotte.lox
```

Si nous voulons une liste des entrées avec les coordonnées :

```
export cave-list -fmt html -location on -o LaGrotte-Entrees.html
```

Au final, notre fichier de configuration *.thconfig* pourrait ressembler à cet exemple [Lien à rajouter].

- **Compiler ?**

Ca y est, nous avons nos fichiers de données formatés pour *Therion*, ainsi qu'un fichier de compilation/configuration qui va nous permettre d'obtenir les exports finaux. Mais maintenant, comment fait-on ?

En fait, il faut transformer en fichier final visuel (par exemple une topographie en pdf) toutes les instructions que nous avons écrites dans nos fichiers ! Il faut donc compiler le projet.

Nous pouvons compiler le projet de deux manières, soit via l'interface graphique *XTherion*, soit via une commande en ligne dans une fenêtre Terminal.

### Compiler avec l'interface graphique *XTherion*

Pour compiler avec l'interface graphique *XTherion*, il nous faut lancer le programme en question. Nous l'avons déjà vu, il s'ouvre en mode compilation.

Il nous faut alors ouvrir le fichier de compilation/configuration *.thconfig* que nous avons écrit. Comme sur la majorité des logiciels, cela se fait en cliquant sur l'icône « ouvrir ». Le *thconfig* est alors chargé dans la fenêtre supérieure. Vous devriez avoir une fenêtre de ce type :

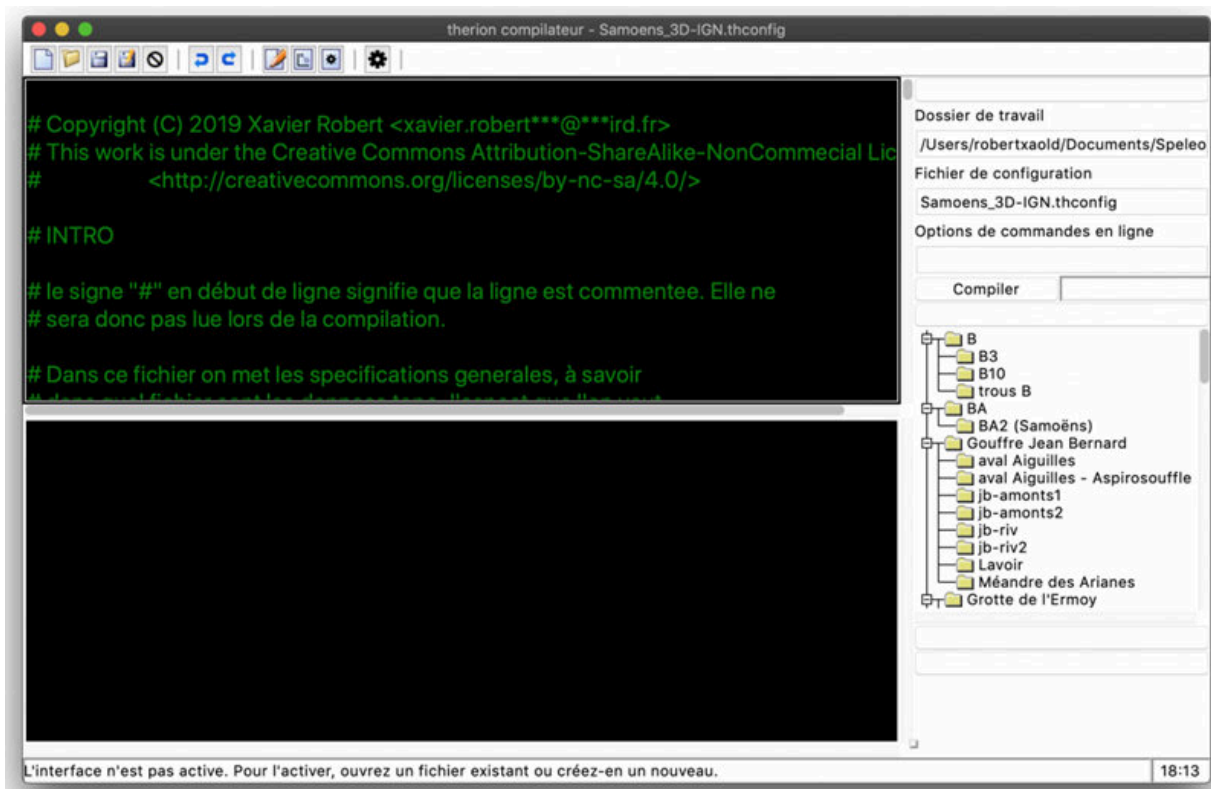


Figure 1 : Le mode compilation de *XTherion* avec un *thconfig* chargé.

Pour lancer la compilation, il suffit de cliquer soit sur la roue dentée dans le bandeau supérieur, soit sur le bouton « compiler » au milieu à droite, dans le panneau de commandes.

Si la topographie est petite, la compilation est rapide. Si la topographie est importante, alors, cela peut prendre plusieurs minutes. Vous voyez l'avancée de la compilation et les résultats des commandes principales dans la fenêtre du bas. Avec la souris, vous pouvez déplacer les ascenseurs de cette fenêtre pour voir toutes les informations (c'est long, il y a beaucoup de choses). C'est ce qu'on appelle le log. Vous devriez obtenir une fenêtre de ce type :

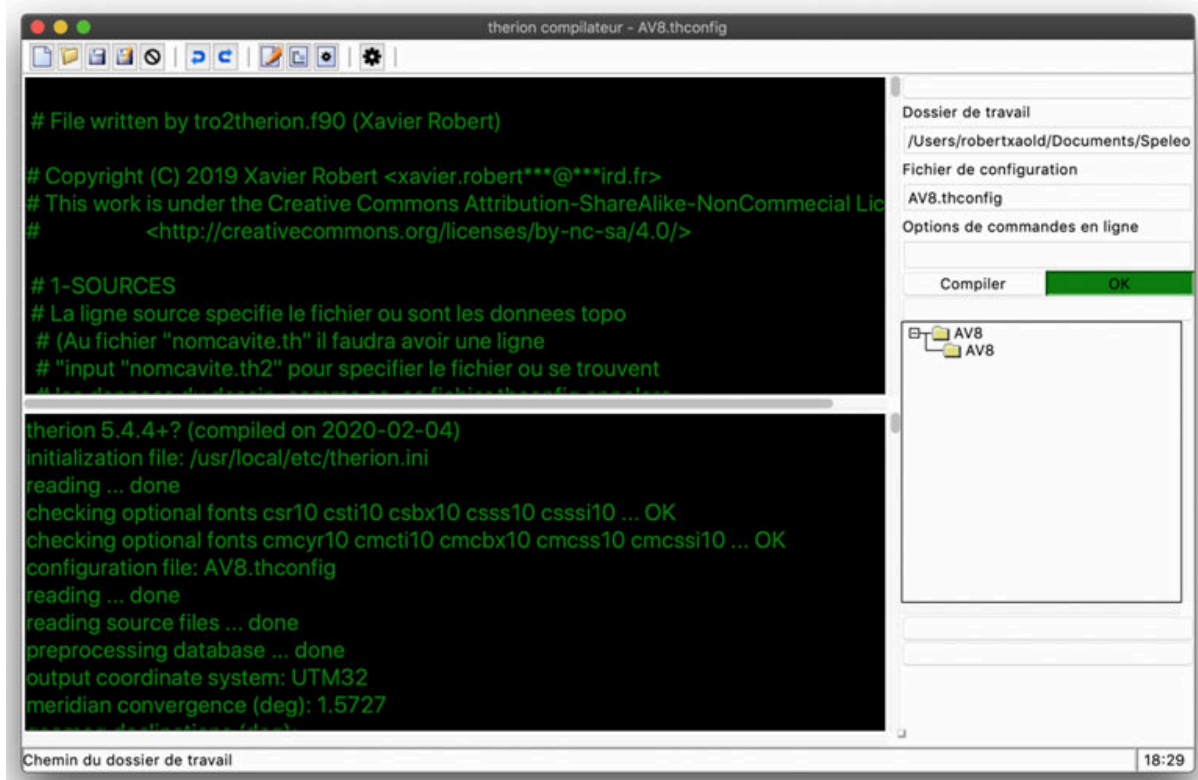


Figure 2 : La fenêtre de XTherion après une compilation réussie

A la fin de la compilation, la case à droite du bouton de compilation change de couleur. Elle devient :

- **Verte** : super, tout va bien, vous devez avoir vos fichiers exportés !
- **Orange** : Aie, il y a un petit problème quelque part (c'est un warning !). Il est probablement sans grande importance parce que *Therion* a réussi à compiler, et vous devriez avoir vos fichiers exportés. Mais la lecture du log devrait pouvoir vous donner des indications quant à votre léger problème.
- **Rouge** : Ouille ! Il y a un problème sérieux, et là, il n'y a pas le choix, *Therion* ne peut pas finir la compilation, donc vous n'aurez pas vos exports. Il faut trouver la cause de l'erreur. Nous verrons plus en détails plus loin comment nous le faisons, mais dans une première approche, le log s'arrête au niveau de l'erreur, avec un message. Si c'est lié à un fichier, vous avez le nom du fichier et le numéro de la ligne problématique. Si vous cliquez sur le numéro de la ligne, ça vous ouvre le fichier en question dans l'éditeur de texte, au niveau de la ligne considérée comme problématique. Après, c'est à vous de jouer...

Avec *XTherion*, ce n'est pas difficile de compiler un projet. Si vous êtes dans une phase de dessin ou d'édition de fichiers dans *XTherion* et que vous voulez compiler pour tester, alors, vous pouvez le faire juste en cliquant sur la petite roue crantée, sans revenir dans la fenêtre de compilation.

### Compiler en utilisant le Terminal

Le Terminal (ou l'invite de commandes sous Windows)... Ca fait peur. Effectivement, nous n'y poussons pas de boutons, nous devons écrire les commandes... Difficile, il faut au moins aligner 2 ou 3 mots clefs... C'est vrai, le plus dur, c'est d'apprendre ces mots clefs. Mais rassurez-vous, pour l'utilisation de *Therion*, ils sont très limités.

Pour de tous petits projets, ce n'est peut-être pas la peine de chercher à comprendre comment ça marche. En revanche, pour de gros projets, où parfois, nous faisons une modification que nous cherchons à répercuter sur un grand nombre de topographies, alors, l'utilisation du Terminal devient, non pas indispensable, mais franchement utile. Nous gagnons un temps fou !



Un fois le Terminal ouvert, il faut aller dans le dossier dans lequel nous voulons travailler et où se trouve le *.thconfig*. Cela se fait avec la commande « cd » (cd pour Change Directory ou Changer de Dossier ; en dessous, « > » indique que c'est une fenêtre Terminal) :

```
> cd le/chemin/vers/mon/dossier
```

Une fois dans le dossier, il suffit de taper la commande :

```
> therion mon_thconfig.thconfigif
```

puis appuyer sur la touche « Entrer »... et attendre sagement la fin de la compilation.

### **Ce que nous obtenons**

Une fois la compilation effectuée avec succès, nous obtenons :

- les exports que nous avons demandés dans le *thconfig*
- un fichier *Therion.log* qui est la copie du log (c'est encore un fichier texte !), ce qui est utile lorsqu'il y a des warnings, ou pour obtenir les statistiques de bouclage. Il est écrasé à chaque compilation

### ▪ **Simplifier les fichiers de configuration ?**

Nous l'avons vu dans la page précédente, un fichier de compilation peut rapidement avoir un nombre de lignes important. Il devient alors un peu fastidieux de chercher ce que nous voulons y modifier. Pour remédier à ce problème, plusieurs solutions sont possibles.

### **Diviser les *thconfig* en fonction de leur caractéristiques**

Si notre *thconfig* est gros parce qu'il permet de gérer la production d'un plan, production d'un atlas, d'une coupe développée et qu'il définit aussi de nouveaux symboles, alors, une première chose qui peut être faite, c'est de faire un fichier *thconfig* spécifique pour le plan, un autre pour la coupe développée, et enfin un dernier pour l'atlas. Dans chacun, nous ne gardons que les blocs layout nécessaire.

Sur les gros projets, ça peut être une première étape qui facilite le travail.

### **Créer un(des) fichier(s) de configuration global(aux) au projet**

Ce point est en fait fondamental si nous travaillons sur de gros projets où nous avons plusieurs *thconfig* qui ont des commandes identiques dans les blocs layout.

Ce que nous pouvons faire, c'est créer un fichier de configuration maître à la racine du projet (c'est à dire dans le dossier le plus haut). Dans ce fichier là, nous allons définir des layouts globaux que nous appellerons dans les layouts des *thconfigs*.

Nous pourrions donner à ce fichier l'extension *.thconfig*, mais pour le différencier des fichiers de compilation, nous pouvons l'appeler par exemple *config.thc* (en fait, le nom des extensions, ce n'est pas si important que ça, du moment que nous savons ce qu'il y a dedans...).

Prenons un exemple. Nous avons un dossier qui correspond à un système karstique avec plusieurs grottes non connectées. Appelons ce dossier « Systeme ». Les données topographiques, les fichiers de compilation et les dessins de chaque grotte sont dans un dossier spécifique pour chaque grotte, et à l'intérieur du dossier « Systeme ». Dans ce dossier système, nous allons avoir par exemple un dossier « grotte1 » qui contient *grotte1.th*, *grotte1.th2* et *grotte1.thconfig*. Chaque grotte a ainsi son dossier.

Pour générer les plans, tous les layouts « plan » des *thconfig* de chaque grotte ont par exemple ces instructions identiques, par exemple pour le fichier *grotte1.thconfig* :

```

source grottel-global.th # contient les données topo et le dessin

layout planG
  language fr
  cs utm32N
  doc-author "Vulcain"
  scale 1/1000
  base-scale 100 m
  ...
endlayout

export map -proj plan -layout planG -o Grottel.pdf

```

Nous pouvons alors copier les instructions qui sont identiques dans un nouveau fichier config.thc qui sera dans le dossier « Système » (donc au dessus de tout le reste). Ce fichier config.thc aura donc la structure :

```

layout planglobal
  language fr
  cs utm32N
  doc-author "Vulcain"
  scale 1/1000
  base-scale 100 m
endlayout

```

Dans chacun des *thconfig* des grottes, nous effaçons (ou commentons dans un premier temps si vous voulez sécuriser le transfert en cas où). Le fichier grotte1.*thconfig* deviendra (il est commenté pour expliquer comment nous faisons) :

```

# Appel des données
source grottel-global.th # contient les données topo et le dessin

# importation du fichier de configuration
input ../config.thc
# ../ signifie que le fichier config.thc est dans le dossier juste au
dessus. S'il fallait remonter de 2 dossiers, nous aurions écrit ../../

layout planG
  # importer les choix globaux. Ils sont dans le layout planglobal du
  # fichier config.thc importé au dessus
  copy planglobal
  #language fr           # Commenté !!!
  #cs utm32N            # Commenté !!!
  #doc-author "Vulcain" # commenté
  #scale 1/1000         # commenté
  #base-scale 100 m     # commenté
  ...
endlayout

export map -proj plan -layout planG -o Grottel.pdf

```

Il suffit de faire ça une fois pour chaque fichier *thconfig* pour chaque grotte. Comme ça, pour tous les changements ou ajouts globaux qui vont s'appliquer à tout le monde, il suffit de modifier uniquement le config.thc.

Créer un config.thc peut vraiment être complexe. Nous avons mis à disposition (sous licence CCby-nc-sa) un tel fichier sur le [template exemple](#).

## 7. FAIRE UN DESSIN AVEC XTherion

### ▪ Faire un dessin avec XTherion

Nous avons vu qu'est-ce que sont les fichiers de données, de compilation et de configuration. Il nous reste à comprendre maintenant comment nous pouvons les utiliser pour dessiner notre topographie.

Nous avons tous nos propres façons d'aborder un dessin topographique :

- Certains ont besoin de dessiner à la main sur du papier sur lequel le squelette topographique a été imprimé.
- D'autres dessinent directement sur l'ordinateur, à la souris, en utilisant le format vectoriel du squelette topographique (pour les utilisateurs de Visual Topo + DAO, ce squelette vectoriel est le fichier .dxf exporté depuis Visual Topo puis importé dans le logiciel de DAO dans le premier calque)

Sous XTherion, ces deux méthodes sont possibles. Nous allons les détailler, et allons voir que finalement, elles ne diffèrent que dans les premières étapes. Cette page est longue, prenez le temps de la lire et de tester par vous même au fur et à mesure, elle contient beaucoup d'informations importantes qui vous seront utiles.

Vous allez le voir, le dessin sous XTherion est d'une grande simplicité en lui même, du moment que nous respectons une succession d'étapes clefs que nous allons décrire dans l'ordre dans lesquelles il faut les effectuer.

La première chose, commune aux deux méthodes, est de lancer le programme XTherion, et d'importer le fichier de compilation que nous allons utiliser (nous en aurons besoin pour pouvoir compiler régulièrement, et ainsi voir l'avancée de notre travail de dessin).

#### **Préparer le terrain : construire le squelette au format vectoriel (xvi)**

Nous avons déjà un peu abordé l'export xvi précédemment. Voici un petit rappel. Pour pouvoir construire le fichier xvi, nous avons besoin d'un fichier de données topographiques (par exemple ici bout-trou1.th) et d'un fichier de compilation MaGrotte.thconfig.

Dans ce fichier, nous pouvons demander un export en xvi sans construire de bloc de mise en page (layout) spécifique, mais nous pouvons tout de même conseiller d'en écrire un de façon simple. Nous pouvons appeler ce bloc pour gérer le xvi, « xviexport ». Au final, il ne va contenir que peu d'instructions, uniquement une indication de l'échelle à laquelle nous voulons dessiner, et la définition d'une grille pour faciliter le dessin. Il aura donc cette structure :

```
source bout-trou1.th # import des données topographiques
#source bout-trou1plan.th2 # import du dessin ; pour le moment commenté
parce que le dessin n'existe pas encore
#source bout-trou1coupe.th2 # import du dessin ; pour le moment commenté
parce que le dessin n'existe pas encore
```

```
layout xviexport
  scale 1 1000 # définition de l'échelle
  grid below # affichage d'une grille en arrière plan
  grid 10 10 10 m # taille de la grille, en mètres. Plus la grille sera
précise, plus le fichier xvi sera lourd
endlayout
```

```
layout MonPlan
  base-scale 1 1000 # la même que celle utilisée pour le xvi
  # Autres instructions pour la mise en page du plan pdf ;
```

```

# voir la page sur la conception d'un fichier de
compilation/configuration
endlayout

layout MaCoupe
base-scale 1 1000 # la même que celle utilisée pour le xvi
# Autres instructions pour la mise en page de la coupe pdf ;
# voir la page sur la conception d'un fichier de
compilation/configuration
endlayout

# export du squelette vectoriel xvi
export map -proj plan -fmt xvi -layout xviexport -output magrotte-
plan.xvi # pour le plan
export map -proj extended -fmt xvi -layout xviexport -output magrotte-
coupe.xvi # pour la coupe

# export des pdf
export map -proj plan -fmt pdf -layout MonPlan -output magrotte-plan.pdf
# pour le plan
export map -proj extended -fmt xvi -layout MaCoupe -output magrotte-
coupe.pdf # pour la coupe

# export du 3D
export model -o magrotte.lox

```

Importons ce fichier de compilation dans *XTherion*, puis compilons une première fois. Cela produit les deux fichiers xvi, le plan et la coupe en pdf, ainsi que le modèle 3D.

Par défaut, sur les pdf, lorsque nous compilons sans importer de dessin dans le fichier de compilation, les seuls éléments présents sont le cartouche, ainsi que la ligne de cheminement (si l'option `symbol-hide centerline` n'est pas activée) avec les stations topographiques. Si dans les données topographiques, il a été déclaré des dimensions gauche, droite, haut, bas, la surface définie par ces dimensions est aussi matérialisée.

Si nous voulons afficher le nom des stations sur le pdf, alors, dans les layout `MonPlan` et `MaCoupe`, il faut rajouter la commande :

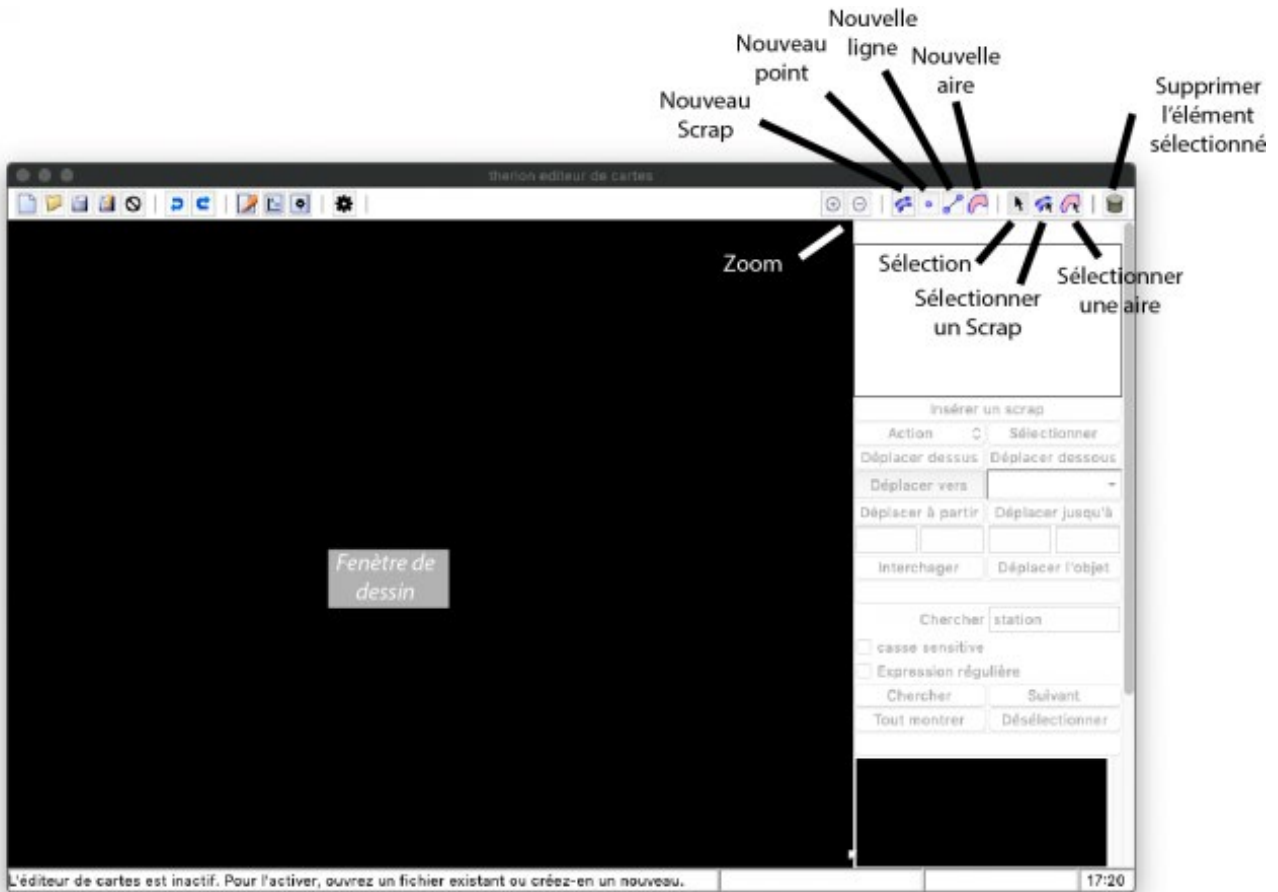
```
debug station-names
```

Si les noms des stations est long et s'il y a beaucoup de stations, cela peut être rapidement illisible. Pour diminuer la taille des stations, une solution temporaire est de jouer dans les layouts `MonPlan` et `MaCoupe` avec l'échelle déclarée pour le dessin (« `base-scale` »). Je vous laisse jouer avec pour vous rendre compte de l'effet. Ceci peut-être très utile si nous souhaitons dessiner sur une impression papier de ce squelette.

### ***Importer le squelette au format vectoriel (xvi) dans l'éditeur de dessin***

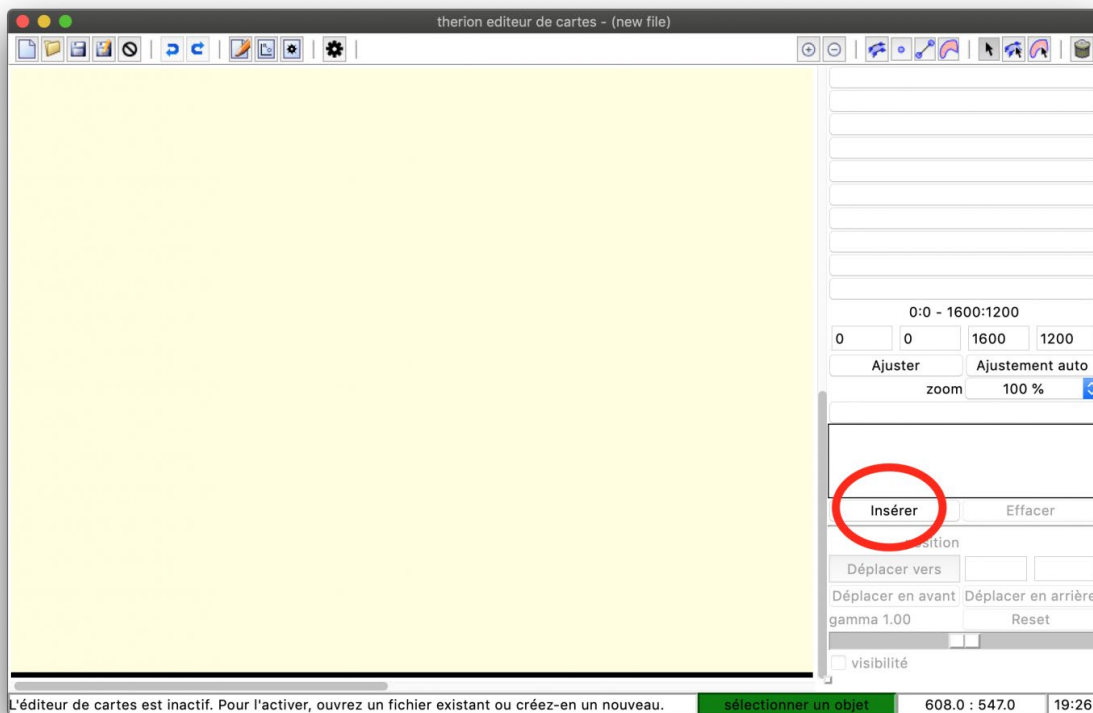
Une fois le fichier xvi exporté, nous pouvons commencer à faire le dessin.

Dans *XTherion*, aller sur l'éditeur de dessin (cliquer sur le bouton correspondant du bandeau supérieur, voir la description du logiciel pour le localiser si ce n'est pas encore fait). Vous allez donc voir cette fenêtre :



### L'éditeur de dessin

Comme indiqué en bas dans la barre d'état, la fenêtre n'est pas active parce qu'il n'y a pas de fichier dessin ouvert. Pour démarrer un nouveau dessin, cliquer sur le bouton « nouveau dessin » . La page devient alors jaune, et les panneaux de contrôles à droite deviennent actifs :



La fenêtre devient active une fois que nous avons créé un nouveau dessin. Noter le bouton « insérer » qui nous permet d'insérer un fichier xvi ou une image de fond

Naviguer dans le panneau de contrôle de droite jusqu'à l'onglet « image d'arrière plan ». C'est normalement le dernier en bas. Ici, cliquer sur le bouton « insérer » .

Le logiciel indique que pour insérer quelque chose, il faut d'abord enregistrer le fichier dessin. Pas le choix, si nous voulons continuer, il faut cliquer sur « OK », et indiquer ensuite où nous voulons enregistrer le fichier et quel nom nous voulons lui donner. Classiquement, nous pouvons lui donner le même nom que le fichier de données topographiques, avec l'indication « plan » ou « coupe » (ex ici : bout-trou1plan.th2 et bout-trou1coupe.th2 respectivement).

Une nouvelle fenêtre s'ouvre, et là, il nous faut choisir le fichier que nous voulons insérer. Choisissons le fichier xvi qui nous intéresse, cela pourrait être ici magrotte-plan.xvi. La fenêtre devient alors :

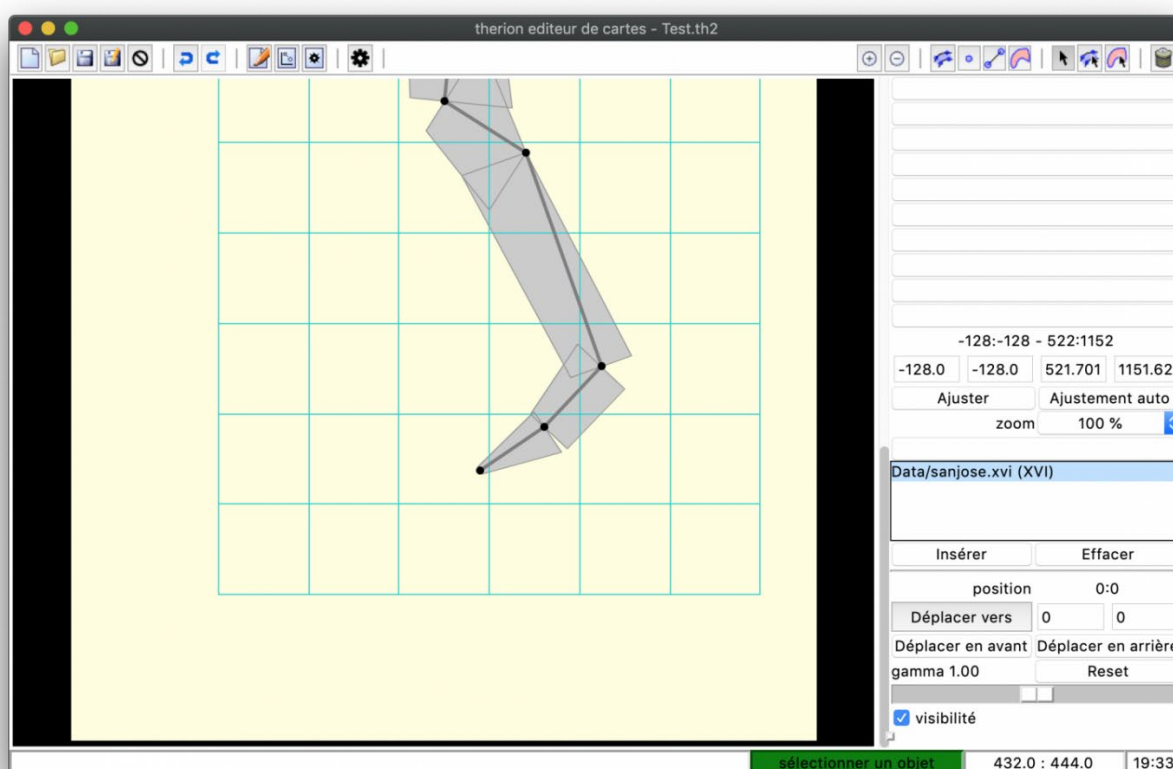


Figure 3 : L'éditeur de dessin après importation d'un fichier xvi.

Nous pouvons jouer avec la position du fichier xvi avec les boutons du bas du panneau de contrôle. Sur la feuille de dessin, noter les points de stations, les dimensions de la galerie et la ligne de cheminement.

Le fond reste jaune, mais maintenant, nous voyons la ligne de cheminement reliant les stations topographiques (points noirs) et à chaque station, le volume de la galerie calculé à partir des dimensions données dans le fichier de données topographique.

Vous pouvez noter que si vous passez votre pointeur de souris :

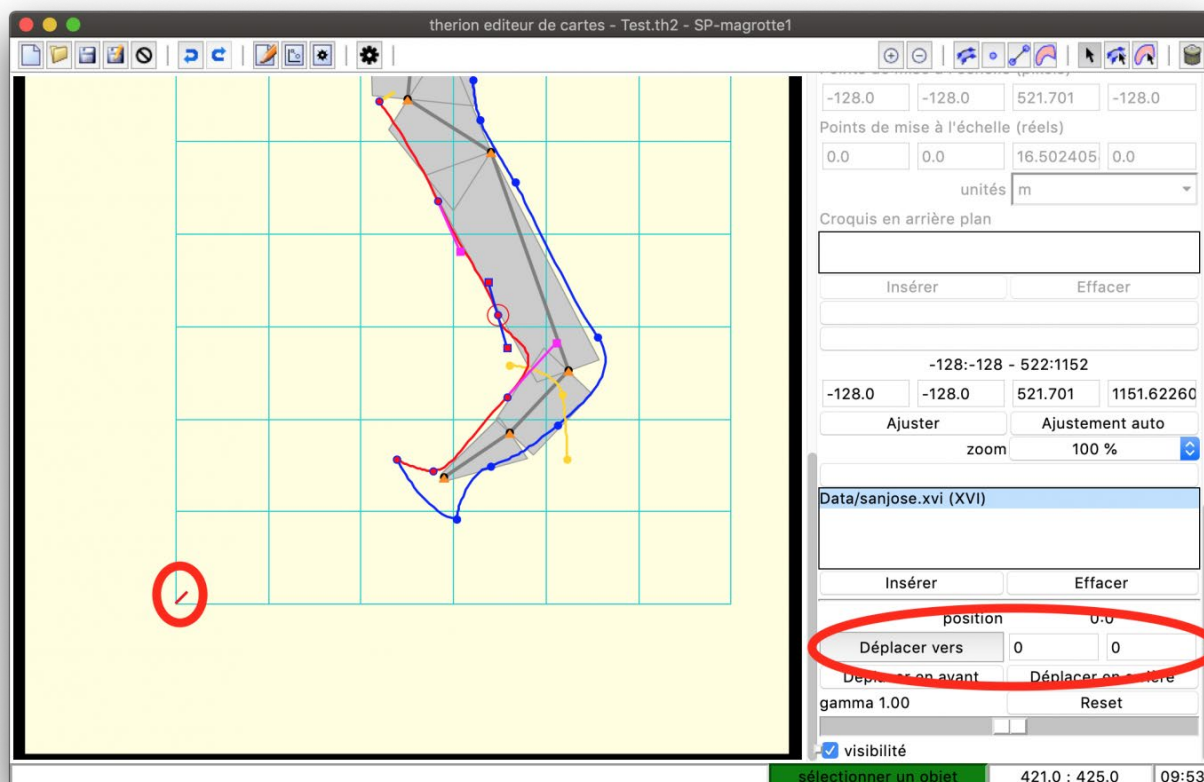
- sur les stations topographiques, le nom de la station apparaît dans la barre d'état en bas ; dans le cas de système complexe, le réseau associé pourra dans certains cas être aussi indiqué ;
- sur la grille, la taille de la grille est indiquée dans la barre d'état en bas ; c'est parfois utile pour pouvoir voir des repères lors du dessin.

En fait, vous le verrez à l'usage, mais dans la barre d'état, vous pourrez lire les informations concernant TOUS les éléments du dessin avec leurs attributs et leurs options du moment que vous les survolez avec le pointeur de la souris. C'est très utile pour se retrouver dans son dessin.

Nous sommes maintenant prêts à vraiment dessiner !

### **Déplacer un fichier de fond**

Lors de l'insertion d'un fichier de fond (xvi ou image), il est possible de le déplacer à notre convenance, en utilisant les commandes du panneau de contrôle, sous la commande d'insertion de fichier. Il faut remplir les deux cases de positions, puis cliquer sur « déplacer vers ». Cela se fait en essai-erreur, c'est assez rapide à prendre en main et à utiliser.



Utiliser les options de déplacement pour déplacer un fichier de fond.

### **Créer son premier scrap (= calque)**

#### **Préambule important**

Une règle de base pour dessiner avec *XTherion*, c'est que nous sommes obligés de dessiner dans des calques, que nous appelons Scraps dans *Therion*.

Ceux qui sont habitués à utiliser Visual Topo et un logiciel de DAO sont habitués au système de calques. Mais, chaque personne a sa propre méthodologie ; certains utilisent un calque par type d'objet dessiné (fond, parois, puits, texte,...), d'autres par zone dessinée (avec pourquoi pas des sous-calques par type d'objet) et même certains, mettent finalement tout dans un même calque sans se soucier d'une quelconque organisation.

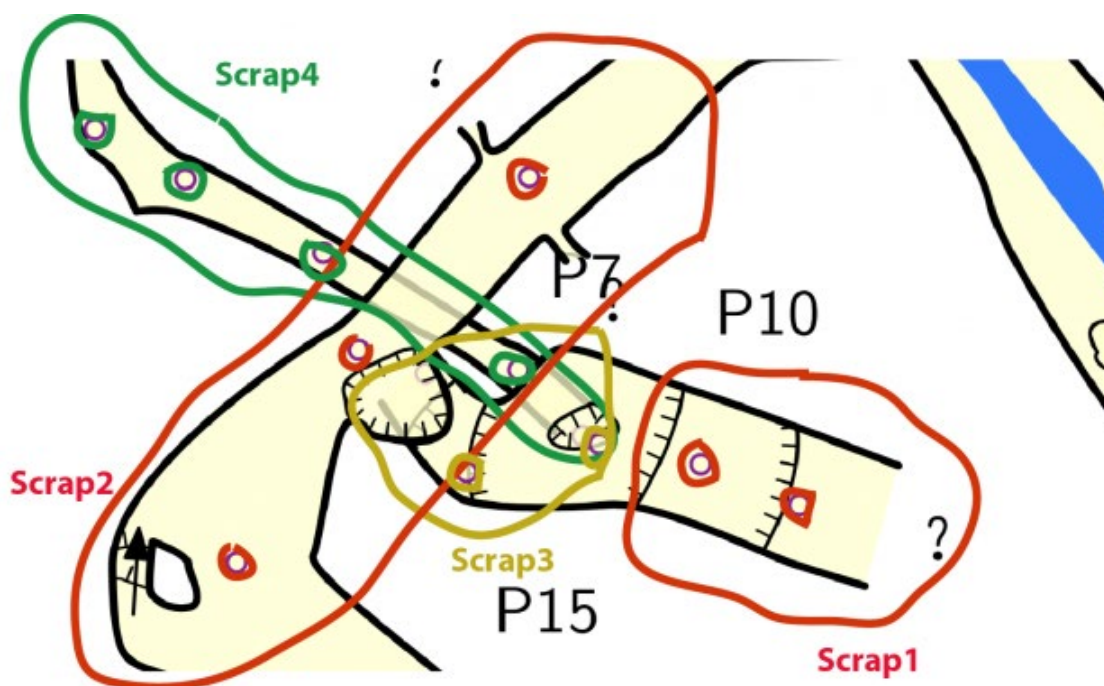
Sous *Therion*, le fonctionnement des Scraps est différent et demande de la rigueur et de la logique. En fait, chaque scrap correspondra à un bout de topographie dessiné complètement (il y a tout, les murs, les stations, le texte,... tous les symboles, en fait !).

Vous allez dire que nous pourrions finalement dessiner toute la topographie sur un seul scrap, ce serait plus simple. Dans certains cas, ce serait possible :

- si aucune galerie ne se superpose :
  - nous allons le voir, s'il y a superposition de galerie, le dessin de chaque bout sur des scraps différents permet à *Therion* de gérer la transparence automatiquement : sur le pdf final, nous verrons la galerie inférieure en transparence ! Et ce, sans s'embêter à le faire manuellement !
  - nous le verrons aussi plus tard, mais les parois des galeries ne doivent pas se croiser !
- et si la topographie n'est pas trop grande : en effet, un scrap énorme sera difficile à exploiter car la réaction de l'éditeur de dessin sera lente voir très lente...

En fait, ces deux points sont finalement deux points qu'il faut avoir en tête lorsque nous prévoyons de créer un nouveau scrap :

- Il doit être petit (idéalement contenir moins d'une trentaine de stations topographiques) ; lors de la compilation, plus les scraps sont grands, plus le risque d'avoir un dessin déformé sur une grande zone est important
- il ne doit pas comporter de galeries se superposant : s'il y a des endroits qui se superposent il faut un scrap par niveau de superposition
- et le corollaire du point précédent, pour permettre la gestion automatique de l'ordre de superposition des scraps et de la transparence sur le dessin final, il est bon d'avoir une idée des altitudes relatives des stations topographiques dans le scrap : par exemple, supposons deux galeries pentues qui se superposent. Il faut que toutes les stations contenues dans le scrap supérieur soient plus hautes que toutes les stations contenues dans le scrap inférieur. Vous le comprendrez rapidement par la pratique, et c'est finalement le point le plus difficile à mettre en œuvre pour faire un beau dessin. Voir la figure ci-dessous :



Voici l'exemple d'une zone complexe avec différents niveaux de superpositions. Ici, idéalement, il faut découper en 4 scraps différents.



Enfin, *Therion* doit connaître le rapport d'échelle du scrap. Ceci peut se faire de deux façon, soit en le définissant graphiquement et en utilisant le panneau de contrôle dédié, soit grâce aux stations topographiques utilisées dans le scrap. Dans ce cas, il faut a minima que le scrap contienne 2 stations topographiques.

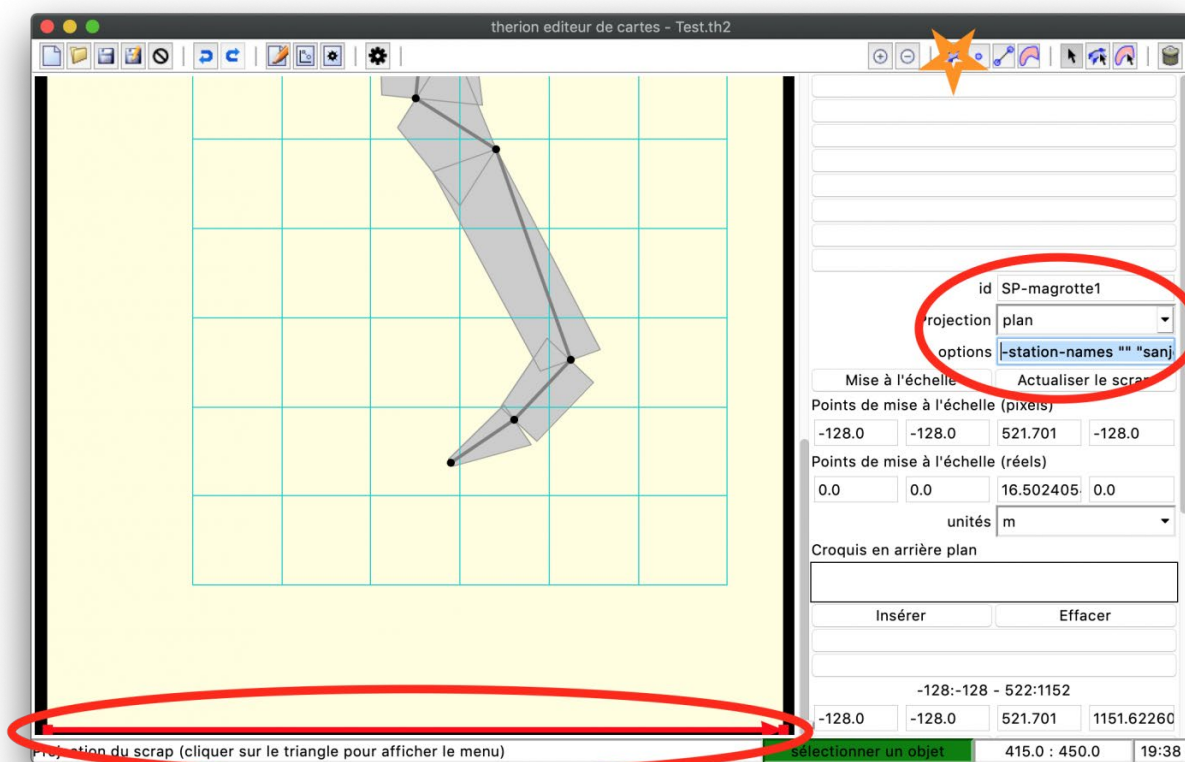
Ce point peut être intéressant à comprendre pour améliorer le lever topographique sous terre : si nous voulons matérialiser correctement un départ dans une galerie ou un puits, où il y a une superposition, alors, il faut qu'il y ait a minima 2 stations topographiques matérialisées dans ce départ... Ce qui souvent manque et peut poser des difficultés (surmontables, mais de façon brouillonne) lors du dessin.

En pratique...

Dans l'éditeur de dessin *XTherion*, une fois le fichier de dessin créé (ou ouvert) et le squelette importé, pour créer un nouveau scrap, il suffit de cliquer une fois sur le bouton nouveau scrap dans le bandeau en haut à droite.

Attention, si vous avez Parkinson est que vous cliquez deux fois de suite, vous créez deux scrap l'un à la suite de l'autre, le premier étant vide. Cela n'a finalement aucune conséquence, sauf à alourdir le fichier de dessin inutilement.

Le panneau de contrôle se positionne alors sur le contrôle des scraps, comme sur l'image suivante :



Création d'un nouveau scrap avec le bouton « Nouveau Scrap » (étoile verte). A droite, les champs important à remplir. En bas, la flèche rouge qui peut servir de mise à l'échelle, si non utilisation des points topographiques (voir la partie ajouter une section de galerie pour plus d'informations).

Dans ce bandeau, il y a plusieurs champs :

- un champ <id> : Ici, nous donnons le nom du scrap. Un conseil (surtout si vous travaillez sur des systèmes complexes), donner un nom qui ait une signification. Par exemple, pour un scrap pour le dessin d'un plan pour la galerie du Petit Dino, nous appellerions le premier scrap de cette galerie « SP-dino-1 », où S veut dire que ce nom correspond au nom d'un scrap, P que c'est pour un plan et 1 que c'est le premier scrap dessiné pour cette galerie. Si nous devons dessiner un second scrap pour cette même galerie, nous l'appellerons « SP-dino-2 ». Pour une coupe développée (C), nous pourrions utiliser la convention « SC-dino-1 », pour une section de galerie, « SS-dino-1 », etc.
- un champ <Projection> : ici, nous définissons le type de projection du scrap que nous dessinons. C'est ce qui permettra à la commande export de savoir quel scrap utiliser pour l'export demandé dans le fichier de compilation. Vous avez le choix entre Plan, Coupe développée, Coupe projetée et None (= aucune). Cette dernière non-projection est utilisée pour dessiner des sections de galeries (voir plus loin)
- un champ <options> : ici, nous écrivons les options du scrap. C'est un des rares champs que nous devons remplir manuellement, et que nous ne pouvons pas remplir par des « clics-clics » ! Ceci-dit, une fois que les options sont définies pour le premier scrap, si nous ne fermons pas le fichier th2, elles seront automatiquement reportées sur les scraps suivants. Il est aussi possible de les indiquer grâce à un copier-coller à partir des options d'un scrap précédent. Les options communément utilisées sont :
  - -author 2020 « Monsieur Vulcain » : c'est l'auteur du dessin, avec l'année de dessin ; il est possible de mettre autant d'auteur que nécessaire,
  - -copyright 2020 « mon CP » : définir un copyright/licence et l'année
  - -stations-names « Préfixe » « Suffixe » : nous y reviendrons dans le paragraphe « ancrer un scrap sur le squelette topographique »

Ces champs sont les plus importants parce que nous les utilisons à chaque fois que nous créons un nouveau scrap.

Les autres champs sont utilisés plus rarement, surtout dans le cas du dessin d'une section, ou du dessin d'une topographie à partir du scan d'un vieux dessin papier. Ils servent à définir l'échelle du scrap dans le cas où nous n'aurons pas suffisamment de stations topographiques pour la définir automatiquement. Nous ne les décrivons pas ici, nous les aborderons dans la partie « dessiner une section ».

Si nous avons plusieurs scraps dans notre dessin, pour naviguer entre les scraps :

- soit nous cliquons sur le bouton « Sélectionner un scrap » du bandeau supérieur à droite
- soit nous utilisons le premier onglet du panneau de contrôle à droite « Objets », qui liste tout les objets (scraps, lignes, points, aires) du fichier de dessin

### ***Ancrer le scrap sur le squelette topographique : définition des stations topographiques sur le dessin***

Un fois un nouveau scrap créé, il faut l'ancrer sur le squelette topographique. Ceci se fait très simplement grâce à l'outil « nouveau point » (bouton à droite, sur le bandeau supérieur).

- cliquer sur le bouton « nouveau point »
- puis cliquer sur la première station topographique que vous voulez dans votre scrap et que vous voyez sur l'arrière-plan xvi
- un petit triangle orange apparaît sur la zone de dessin (c'est le point « station topographique », et le panneau de contrôle se positionne sur l'onglet « Points ». Automatiquement, le type de point est sur « station topo » et la case option contient le texte « -name 1 » si votre station est la station 1 dans le fichier de données.

- pour ajouter une seconde station cliquez sur une nouvelle station sur le xvi, et regardez la case option. Maintenant, vous avez le texte « -name 2 » si votre station est 2 dans le fichier de données. Quand nous faisons ces étapes à partir d'un xvi, le nom de la station dans la case option est automatique, c'est très pratique !
- Faire ceci pour toutes les stations qui seront contenues dans ce scrap.

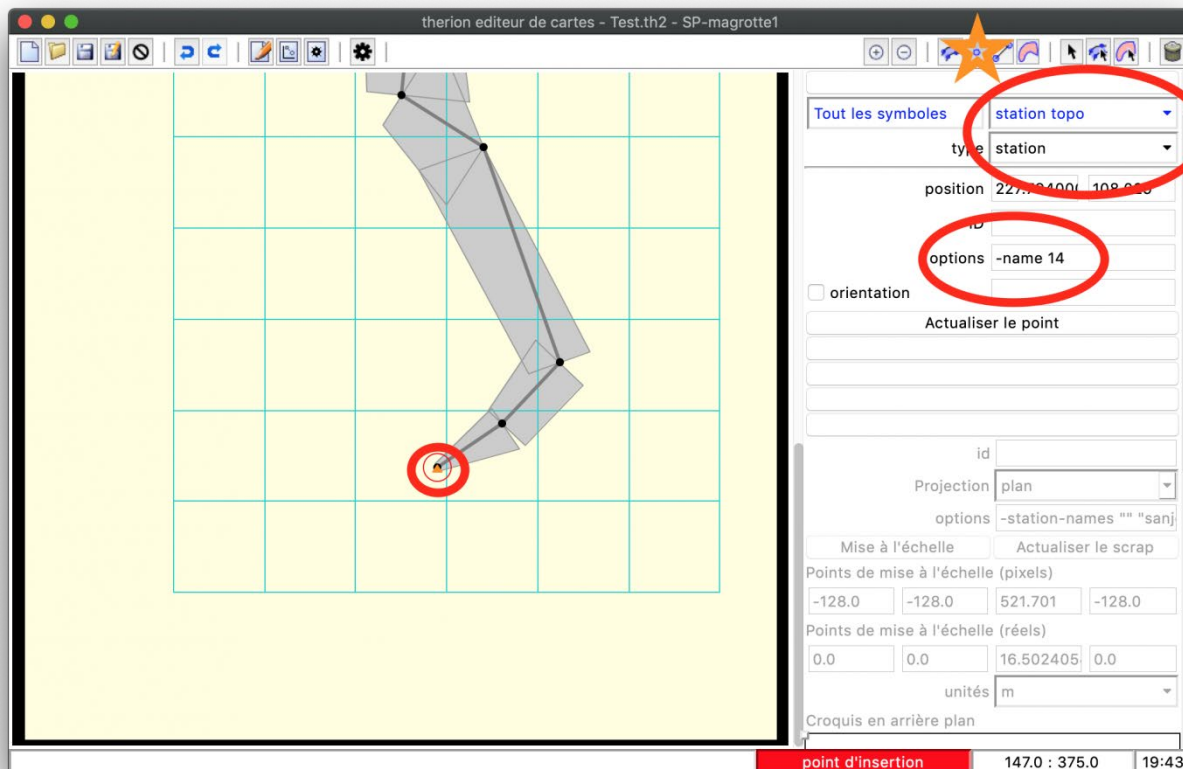


Figure 4 : Insertion du premier point Station topographique

L'insertion du premier point Station topographique grâce à l'appui sur le bouton nouveau point (étoile orange). Dans le panneau de droite, la sélection du type de point, et les options du point.

Pour déplacer un point, cliquer sur le point, il devient avec un cercle rouge autour, puis déplacer-le où vous le voulez.

Il arrive que nous fassions des erreurs lors de cette étape. Il peut être utile d'effectuer une vérification que tout va bien. Pour cela :

- dans le fichier de configuration *thcongif*, décommenter la ligne d'importation du dessin (passer dans la fenêtre de compilation pour cela)
- revenir dans l'éditeur de dessin
- dessiner les deux premières stations du scrap
- compiler (cliquer sur la roue crantée du bandeau supérieur)

Si avec, un exemple simple, vous avez suivi exactement les étapes ci-dessus, alors, la fenêtre de *XTherion* passe automatiquement sur la fenêtre compilation et le logiciel vous dit qu'il y a une erreur, comme quoi la station 1 (ou 2) n'existe pas. Le message d'erreur est du type « A COMPLETER »

La raison, c'est que nous avons bien défini le nom des stations dans le scrap (avec l'option « -name 1 »), mais en fait, nous n'avons pas dit à quelle topographie (survey) cette station appartient. Pour une petite cavité qui contient un seul fichier de données topographiques, ceci

parait peu compréhensible, mais cela devient finalement la seule solution pour les cavités complexes comprenant plusieurs topographies. Comme les noms des stations topographiques peuvent être redondants entre les différentes topos, il nous faut pouvoir les différencier (dire à quelle topographie appartient la station), et c'est là qu'entre en jeu l'importance du nom de la topo (rappelez vous, l'identifiant écrit après la commande survey dans le fichier de données).

Reprenons l'exemple de notre grotte avec le fichier topographique `bout-trou1.th` contenant la `survey matopo`. Dans la case option du point « station topo », nous indiquerons que la station 1 appartient à la `survey MaTopo` en écrivant dans la case option

```
-name 1@matopo
```

Si la `survey MaTopo` est incorporée dans une `survey globale` (rappelez-vous le paragraphe sur la jonction de différentes `surveys`), nous pourrions écrire dans certains cas :

```
-name 1@matopo.globale
```

et ainsi de suite...

Pour définir les stations topographiques dans un scrap, il faut faire cela pour toutes les stations qui appartiennent au scrap. Cela peut-être rébarbatif...

Il est possible de simplifier l'opération, directement à partir de la définition du scrap, grâce à l'option du scrap `station-names`. Pour cela, il faut utiliser l'outil de sélection de scrap pour sélectionner le scrap sur lequel nous travaillons. Dans les options de ce scrap, il faut rajouter :

```
-station-names "" "@matopo"
```

En fait, nous laissons le préfixe vide, et imposons un suffixe identique à tous les noms de stations topographiques. Comme ça, plus besoin de le faire manuellement pour chaque station topographique, nous gagnons un temps certain !

### ***Dessiner le contour de(s) galerie(s) : la ligne mur/wall***

Une fois le scrap ancré sur le squelette topographique (grâce au dessin des points stations-topo), l'étape suivante est de dessiner le contour des parois.

En effet, sous *Therion*, le dessin du contour des parois va déterminer ce qui appartient à l'intérieur de la cavité, et ce qui est à l'extérieur. Cela aura une conséquence :

- sur le remplissage couleur de l'intérieur de la cavité
- sur l'affichage ou nom des autres symboles que nous allons dessiner (et qui sont généralement visible qu'à l'intérieur d'une cavité, ce qui est très pratique pour dessiner !)

Une paroi, c'est une ligne. Nous allons donc les dessiner grâce à l'outil « nouvelle ligne ».

- Dessiner une ligne

Pour cela, après avoir dessiné les stations topographiques, cliquer une fois sur le bouton « nouvelle ligne » dans le bandeau supérieur à droite. Quand on clique sur ce bouton, cela crée automatiquement une nouvelle ligne.

Donc, comme pour les scraps, si vous cliquez plusieurs fois, vous créez plusieurs lignes vides. Ce ne change rien pour la compilation, mais ça alourdit inutilement le fichier de dessin !

Une fois la nouvelle ligne créée, il faut commencer à dessiner

- Cliquons une fois sur le dessin, où nous voulons démarrer notre ligne de paroi ;
- Un point apparaît, c'est le début de notre ligne ; Le panneau de contrôle à droite se positionne sur l'onglet ligne, où nous pouvons visualiser le type de ligne que nous dessinons (il y a une case pour le nom en français et une case pour le nom équivalent en anglais). Si nous travaillons sur un nouveau document, la première ligne créée est automatiquement une ligne de paroi (mur/wall). Sinon, le type de la ligne sera le dernier type utilisé. Il faut utiliser les ascenseurs pour sélectionner le type « mur/wall ». Pensez à le vérifier à chaque nouveau scrap, sinon, aucune paroi ne s'affichera sur le dessin final.
- Cliquons ensuite où nous voulons notre deuxième point de ligne. Si nous cliquons et relâchons le clic rapidement, alors, la ligne sera une ligne anguleuse, ce qui sera peu esthétique. Si nous voulons adoucir les angles, lors de la création du point de ligne, il faut appuyer longuement et en même temps déplacer le pointeur. Nous verrons alors deux tirettes apparaître, elles nous permettent de gérer l'adoucissement de l'angle. Attention à ne pas faire faire de tour sur lui même parce que ça va générer une erreur.
- Continuons ainsi de point en point pour construire la ligne. Il ne sert à rien de faire trop de points, il faut dessiner uniquement les points nécessaires (ça vient très rapidement avec la pratique)

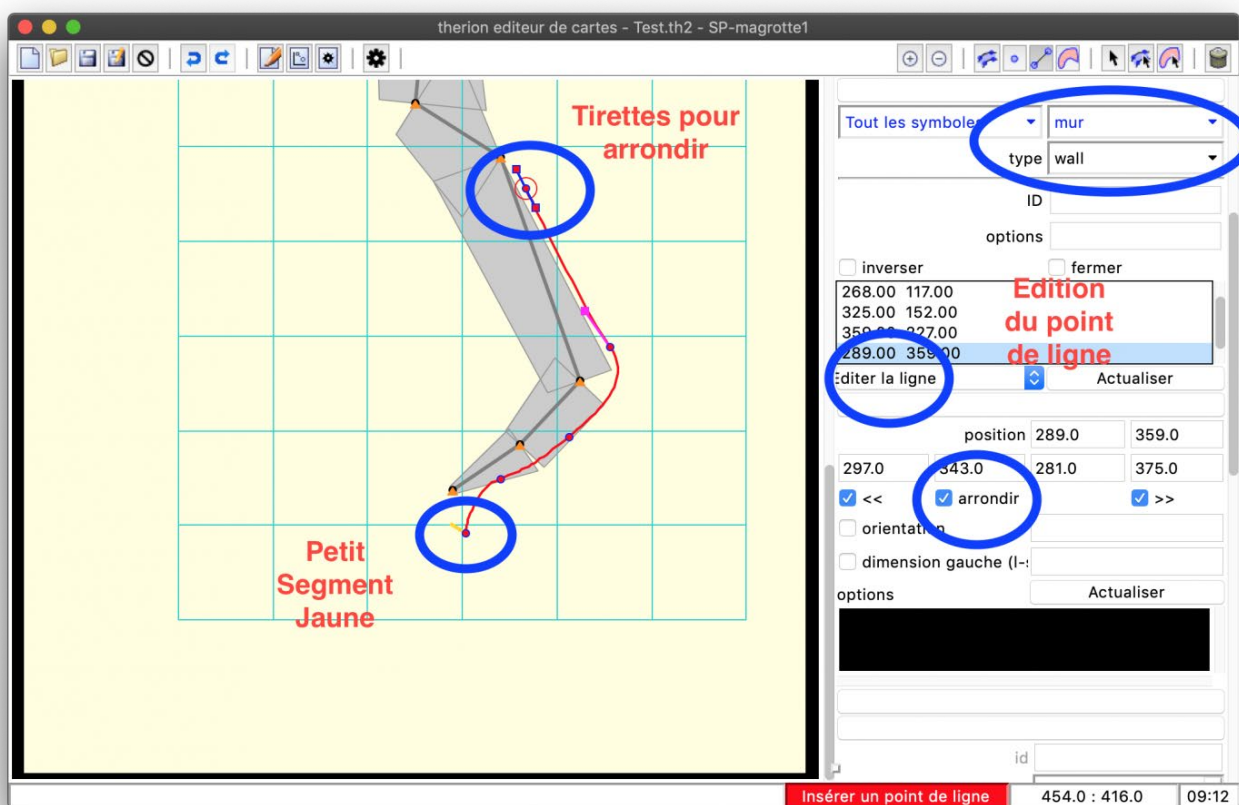


Figure 5 : Dessin de la première ligne « Paroi (wall) ».

Noter le type, mais aussi le petit segment jaune marquant l'intérieur de la galerie. Le dernier point en haut est le point actif, avec les deux tirettes pour arrondir l'angle.

- Orienter une ligne

Dans *Therion*, une ligne n'est pas qu'un trait simple. C'est un trait qui est orienté. Lorsque nous commençons à dessiner une ligne, au niveau du point de départ de la ligne nous pouvons remarquer un tout petit segment jaune. Ce petit segment jaune est très important parce qu'il marque le côté de la ligne qui va définir l'intérieur de la grotte. Généralement, il est à gauche dans le sens du dessin de la ligne.

S'il n'est pas positionné du bon côté, pour le repositionner correctement, il suffit, dans l'onglet ligne du panneau de contrôle, de cliquer sur « Inverser ». Le segment jaune changera de côté.

### ***Editer une ligne***

Dans l'éditeur de dessin, la ligne active est rouge, et les autres sont bleues, verte, grises,...

Pour sortir de l'édition de la ligne, il suffit de cliquer sur la petite flèche noire dans le bandeau supérieur à droite (« Sélection »).

Pour déplacer un point de la ligne, cliquer sur le point, puis déplacer-le où vous le voulez.

Pour effacer un point, sélectionner le point à effacer, puis dans l'onglet ligne du panneau de contrôle, cliquer sur Editer la ligne → effacer le point.

En fait, le menu « Editer la ligne » est utile car il permet :

- d'effacer un point de ligne,
- de couper une ligne
- de rajouter un point de ligne

Pour jonctionner deux lignes, il suffit de créer une nouvelle ligne, et de la commencer (ou la finir) en cliquant sur le point de la ligne précédente ou nos voulons faire la jonction. Ceci fonctionne aussi entre deux lignes appartenant à des scraps différents.

### ***Vérifier le travail***

Pour vérifier que c'est bien fait, une fois les parois dessinées, compiler et vérifiez sur votre pdf le résultat.

- Cas d'une paroi non fermée et qui s'ouvre vers l'extérieur

Ceci peut arriver au niveau d'une entrée par exemple : la ligne de paroi ne sera pas fermée pour matérialiser l'entrée, et si les parois de l'entrée sont en entonnoir et/ou s'ouvrent vers l'extérieur, alors, la compilation risque de générer un warning et, les exports kml peuvent ne pas afficher le scrap en question.

Pour y remédier, il suffit de fermer la zone d'entrée (c'est à dire de relier le bout de chaque ligne paroi formant la zone d'entrée) avec une nouvelle ligne « paroi (wall) », avec le petit segment jaune vers l'intérieur de la cavité.

Le problème, c'est que nous ne voulons pas que cette paroi soit visible. Pour cela, sélectionner la ligne de jonction (outils « Sélection »), puis faire un clic droit sur cette ligne. Aller dans sous-type – > invisible. Compiler. Le tout est joué !

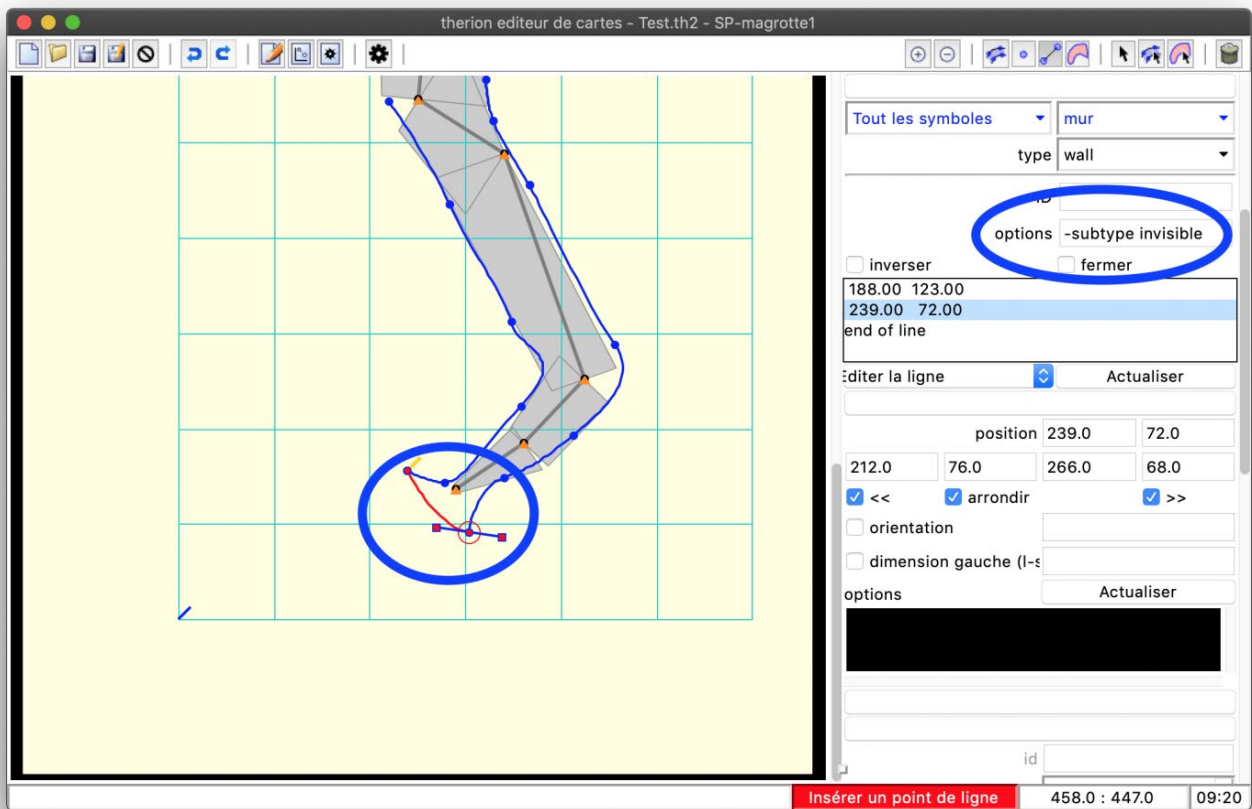
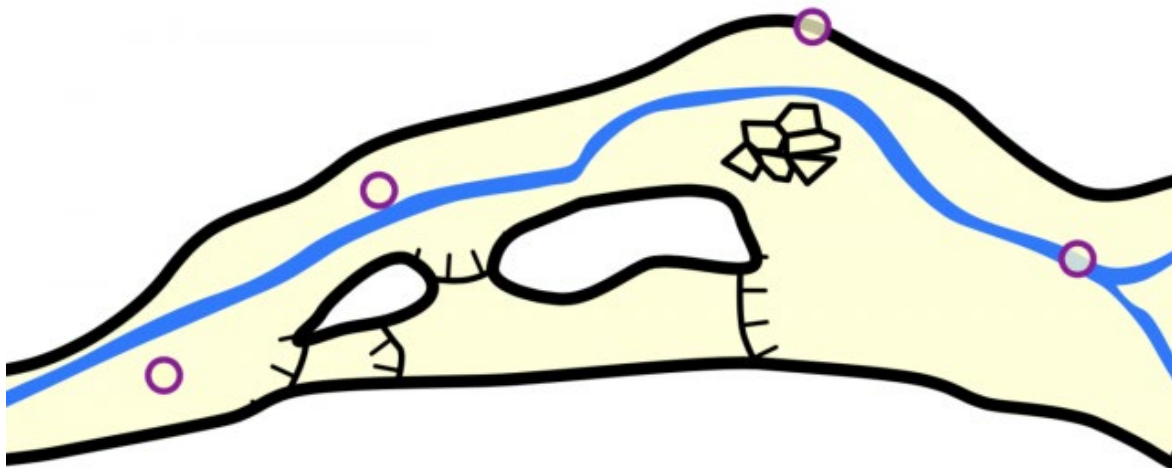


Figure 6 : Exemple du dessin d'une entrée, avec les deux lignes de parois ouvertes vers l'extérieur.

Pour éviter un warning, nous fermons la zone d'entrée avec une nouvelle ligne de paroi, avec l'option invisible. Comme ça, nous délimitons bien l'intérieur de la cavité, mais ne voyons pas la paroi sur le dessin final. Noter que la jonction des lignes se fait simplement par superposition des points de ligne des extrémités (facilité par l'attraction des nouveaux points par les points existants).

- Cas des piliers (bouclage autour une paroi à l'intérieur de la galerie)

Ici, nous définissons le terme de « pilier » comme étant définie par la paroi interne d'un bouclage, comme pour l'image ci-dessous :



Exemples de piliers. Ici, il y a deux piliers (les zones blanches à l'intérieur de la galerie principale).

Si une partie de ce bouclage est sur un scrap, et une autre partie sur un autre scrap, alors, il n'y a pas de problème, nous n'avons pas besoin de faire quoi que ce soit de plus.

En revanche, si ce pilier est complètement à l'intérieur d'un même scrap, cela pose problème. En effet, nous avons montré que l'orientation de la ligne paroi définit ce qui est à l'intérieur et ce qui est à l'extérieur de la cavité. La ligne paroi du pilier indique donc une zone à l'extérieur de la cavité. Le problème, c'est que la ligne paroi de la galerie (qui ne définit pas le pilier) définit aussi un intérieur et un extérieur. Au final, nous nous retrouvons avec un extérieur à l'intérieur d'une galerie. Vous avez compris ? Non ? Et bien *Therion*, c'est pareil, il ne comprend pas...

Pour palier à ce problème, il faut donc dire à *Therion* que la ligne paroi qui définit le pilier et à l'intérieur de la grotte. Pour cela :

- Sélectionner la ligne paroi formant le pilier
- faire un clic droit sur la ligne et aller à « autres options → contours (outline) → à l'intérieur (in) » ; Par défaut, cette option est « à l'extérieur (out) ». La valeur « aucun (none) » permet de dessiner une ligne paroi sans définir d'intérieur ou d'extérieur.
- Dans la case option, nous voyons alors apparaître le texte « -outline in »
- Si le pilier est défini par plusieurs lignes mur, il faut faire ceci pour chaque ligne paroi du pilier
- Attention, la prochaine ligne suivant aura cette option, il faudra penser à l'effacer si la prochaine ligne ne concerne pas un pilier !

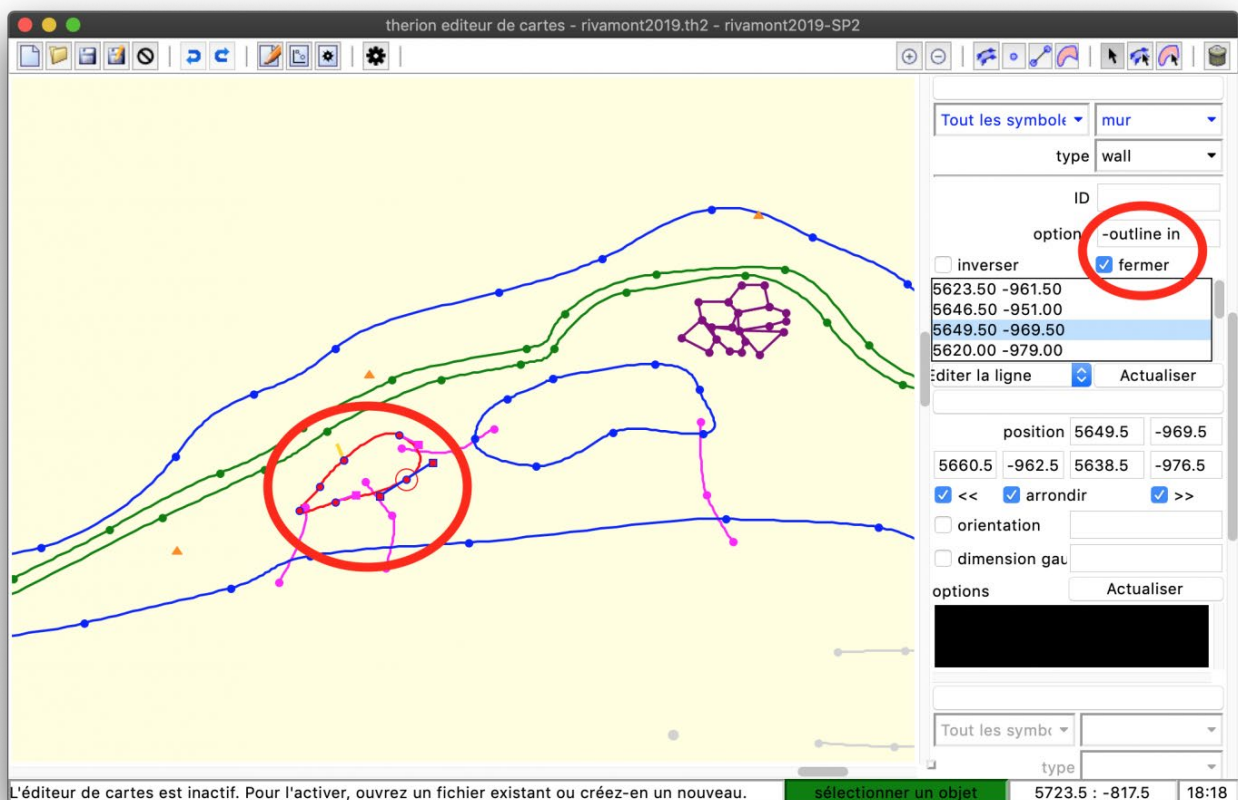


Figure 7 : Le dessin d'un pilier

Pour dessiner un pilier, noter l'option « -outline in » qui permet de dire que c'est une paroi formant un pilier à l'intérieur de la galerie.



## **Ajouter les autres informations sur la topographie**

Une fois que les points de stations topographiques et les lignes de parois/mur/wall ont été dessinés le plus gros est fait. Il ne reste plus qu'à dessiner l'habillage, c'est à dire les autres points, lignes et aires, sur le même modèle !

### **DESSINER EN UTILISANT UN DESSIN EFFECTUÉ A LA MAIN**

#### **Préparer le squelette topographique à imprimer**

Le plus facile, c'est d'imprimer un pdf avec le squelette topographique et le nom des stations. Pour cela, dans le fichier de compilation, il faut utiliser un layout qui contiendra a minima ces commandes :

```
scale 1 1000 # A Adapter en fonction de votre topographie/desiderata
debug station-names # pour afficher les noms de station
```

Ensuite, soit nous exportons le plan et la coupe classiquement, soit nous exportons un atlas pour le plan, ce qui permet de travailler sur des pages au format A4, avec la commande d'export

```
export atlas -layout MonLayout -o magrotte-atlas.pdf
```

Imprimer, dessiner, puis scanner.

#### **Importer le scan du dessin à la main dans XTherion**

Une fois le scan effectué, il faut enregistrer le dessin scanné dans un format importable dans XTherion. Le format gif va très bien.

Il faut ensuite suivre les mêmes étapes que pour importer un xvi :

- ouvrir XTherion
- aller dans l'éditeur de dessin
- Ouvrir un nouveau fichier de dessin
- Dans le panneau de contrôle, aller dans l'onglet « image d'arrière-plan » et cliquer sur « insérer »
- enregistrer le fichier th2
- importer le dessin gif scanné

le dessin scanné apparaît alors sur le plan de dessin. Vous êtes prêts à passer à la suite....

#### **Numériser le dessin sous XTherion**

En fait, ce sont les mêmes étapes que précédemment, mais au lieu de se baser sur un xvi, nous nous basons sur le dessin scanné :

- créer un nouveau scrap
- ancrer le dessin sur le squelette topo en dessinant les stations topographiques sur les stations topographiques du scan en gif. Attention, ici, le dessin scanné ne contient pas les informations du squelette topographique, et donc, il faut manuellement entrer le nom de la station topographique pour chaque point, en écrivant dans la case option « -name <nom\_station> ». C'est pour ça qu'il est intéressant d'avoir le nom des stations sur le dessin scanné, cela facilite ce travail !
- dessiner les parois (ligne mur/wall)
- et dessiner le reste !

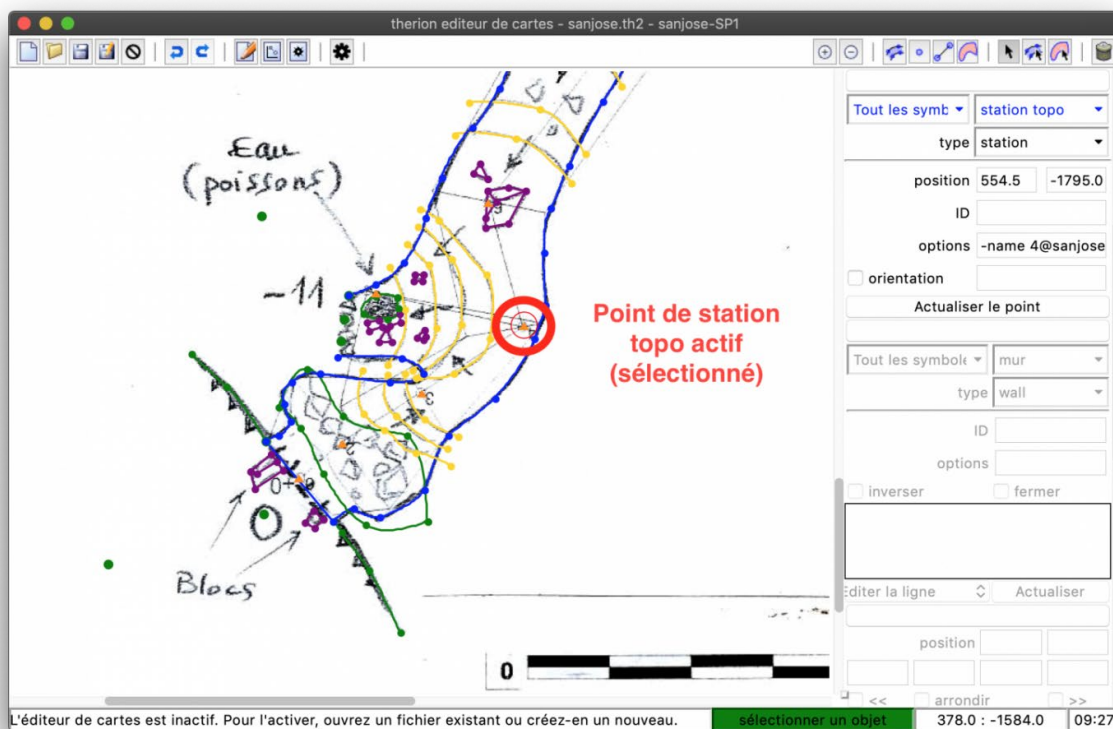


Figure 8 : Dessin à partir d'un dessin scanné.

Le scrap Sanjose-SP1 (le nom est dans le titre de la fenêtre, ça peut -être utile à savoir) est mis à l'échelle par le dessin de points « station topo ». Nous pouvons dessiner ainsi trait à trait le dessin fait à la main.

Et voila, votre scrap est prêt, il ne reste plus qu'à compiler pour obtenir un beau résultat !

Avec *XTherion*, pour dessiner, nous n'allons manipuler que deux entités différentes, les points et les lignes. Pour modifier le figuré final, il suffit de jouer avec les types des entités et leurs options. Nous verrons plus loin que pour les aires, finalement, ce n'est qu'un travail à partir de lignes.

Nous avons décrit dans « Faire un dessin avec *XTherion* » comment insérer un point « station topographique ». Pour insérer un autre type de point, il suffit de faire pareil (évidemment, toujours dans un scrap, sinon la compilation va planter sur une erreur...) et de changer le type du point. Nous pouvons aussi le faire par un clic droit sur le point en question. Ainsi, nous pouvons aussi modifier le sous-type du point s'il possède des sous-types.

Ci-dessous, nous décrivons les principaux types de point utilisés classiquement mais un peu particuliers, et leurs options utiles. Commençons par quelques options communes à la plupart des points.

### **Options communes à la plupart des points**

Les options décrites ici s'inscrivent dans la case option du point correspondant (dans l'onglet « Point » du panneau de contrôle latéral), ou sont accessible par un clic droit sur le point. Les options de visibilité et de positionnement s'appliquent aussi pour les lignes et pour les aires.

#### **Taille du figuré : l'option `-scale <xs, s, m, l, xl>`**

Par défaut, le figuré du point sera de taille M. Si nous voulons le modifier, il faut utiliser l'option :

`-scale taille`

ou taille peut être, du plus petit au plus grand : xs, s, m, l ou xl.

Une valeur numérique peut aussi être utilisée.

### **Afficher un point à l'extérieur de la galerie, ou cacher la partie du figuré à l'extérieur de la galerie : l'option -clip <off/on>**

Par défaut, certains points ne vont pas s'afficher à l'extérieur de la galerie dessinée. D'autres, en revanche vont s'afficher à l'intérieur et à l'extérieur de la galerie dessinée. Si nous souhaitons modifier ce comportement pour un point, il faut modifier la valeur de l'option :

```
-clip on # Le figuré du point ne sera pas affiché à l'extérieur de la galerie dessinée,  
-clip off # Le figuré du point sera affiché à l'extérieur de la galerie dessinée,
```

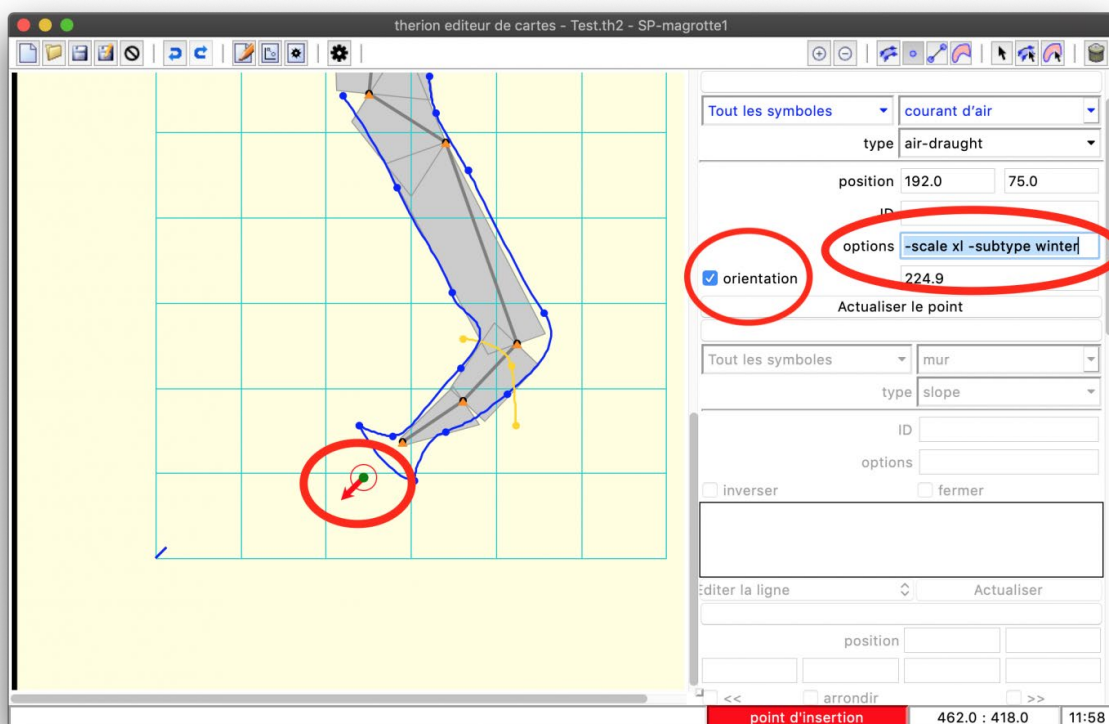
Cette option est aussi valable pour les lignes et les aires.

### **Rotation/orientation d'un point**

Tous les points ne peuvent pas être orientés. C'est le cas du point station topographique par exemple. Mais beaucoup le peuvent.

Une fois le point inséré dans le scrap, le sélectionner, puis dans l'onglet « Point » du panneau latéral, cocher la case « orientation ». Sur la partie dessin, une flèche rouge apparaît. Ensuite, pour orienter le point :

- soit entrer l'angle d'orientation dans la case prévue à cette effet dans le panneau latéral,
- soit passer le pointeur de la souris sur la flèche du point que nous voulons orienter. Elle devient jaune. Cliquer dessus permet, en laissant appuyé, d'orienter le symbole visuellement.



Un point (ici Courant d'air (air-draught)) orienté. Ici, le symbole indiquera que c'est soufflant (sort de l'entrée). Noter la case cochée, et la flèche. Ce point possède aussi les options `-scale xl` (ici) et `-subtype winter` pour dire que le sens de courant d'air ainsi dessiné est valable en hiver.

### Visibilité d'un point

Les points sont généralement visibles. Pour rendre un point invisible, activer l'option.

```
-visibiliy off
```

Pour activer la visibilité, il suffit de remplacer off par on.

### Placement relatif d'un point

Il se peut que par défaut, le point dessiné ne soit pas placé sur un niveau qui nous convient, par exemple en dessous d'une aire, alors que nous le voulons au dessus. Nous pouvons jouer avec ce positionnement grâce à l'option :

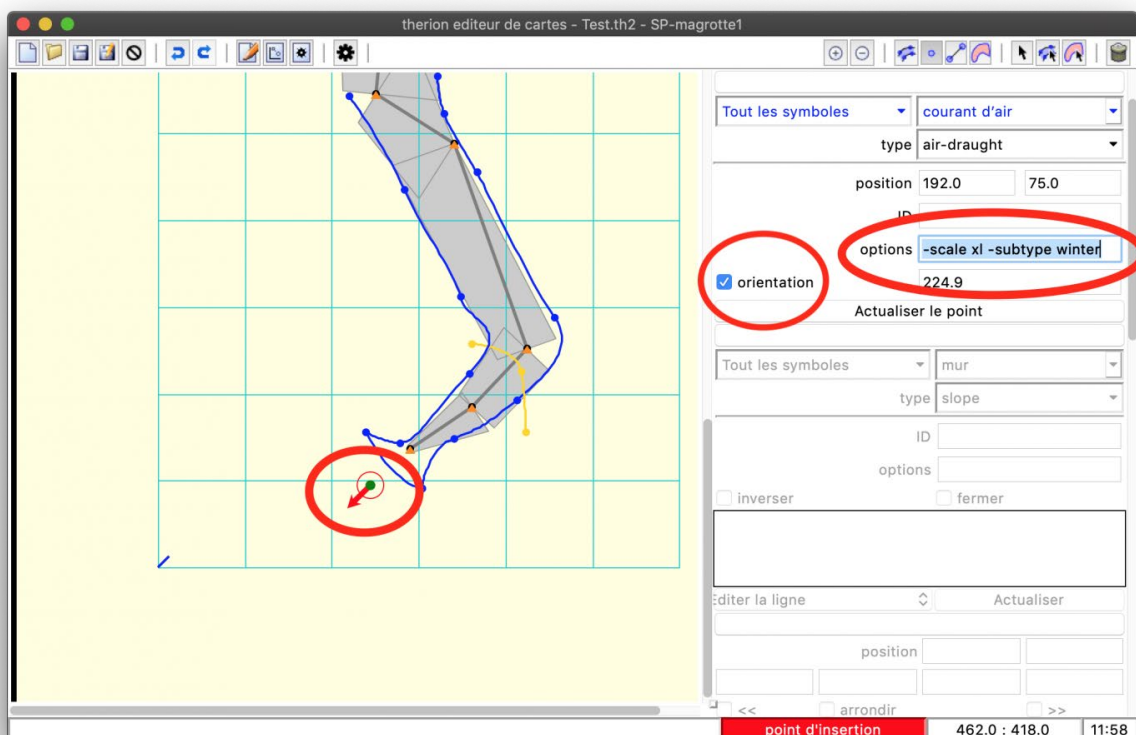
```
-place <top/bottom>
```

« Place » veut dire placer, et si la valeur est « top » (au dessus), alors, le point sera place au dessus de tout le reste du scrap. Inversement, si la valeur de l'option est « bottom », alors, le point sera en dessous.

### Le point Courant d'air (air-draught)

Il est simple, mais il peut être orienté, et prendre le sous-type « summer » (été) ou « winter » (hiver) pour indiquer le sens du courant d'air en été et en hiver si nous le savons.

Il faut aussi noter que l'option `-scale` va jouer sur le nombre de barbules de la flèche de courant d'air.



Un point (ici Courant d'air (air-draught) orienté. Ici, le symbole indiquera que c'est soufflant (sort de l'entrée). Noter la case cochée, et la flèche. Ce point possède aussi les options `-scale` (`xl` = beaucoup de barbules sur la flèche, ici) et `-subtype winter` pour dire que le sens de courant d'air ainsi dessiné est valable en hiver.

### ***Le point altitude (altitude)***

Ce point sert à indiquer l'altitude d'un point donné. Il est sous la forme d'un petit point situé à l'endroit où nous avons positionné le point sur le dessin, auquel est rajoutée une étiquette indiquant l'altitude.

### ***Position de l'étiquette d'altitude***

Si nous ne faisons rien, le libellé de l'étiquette est ajouté sur le point, au milieu. Ce n'est pas beau parce que ça recouvre le point.

Nous pouvons imposer le positionnement de l'étiquette avec l'option

```
-align <valeur>
```

Cette option peut prendre la valeur :

- `top` (= haut en français) : l'étiquette sera centrée sur le point, mais au dessus du point
- `bottom` (= bas en français) : l'étiquette sera centrée sur le point, mais au dessous du point
- `left` (= gauche en français) : l'étiquette sera à gauche du point
- `right` (= droit en français) : l'étiquette sera à droite du point
- et les combinaisons `top-left`, `top-right`, `bottom-left` et `bottom-right` pour que l'étiquette soit respectivement dans les coins en haut à gauche, en haut à droite, en bas à gauche ou en bas à droite du point.

Nous pouvons donner cette valeur à l'option en l'écrivant dans la case « option » du point, ou graphiquement avec un clic-droit sur le point et en allant dans « aligner ».

### ***Valeur de l'étiquette d'altitude***

Sur le dessin compilé, ce point pourra avoir plusieurs valeurs :

- Si dans les données topographiques, aucune coordonnée d'entrée n'est donnée, alors, la valeur indiquée par l'étiquette sera le dénivelé de la station la plus proche par rapport à l'entrée.
- Si dans les données topographiques, les coordonnées de l'entrée sont données (et donc avec la valeur de l'altitude d'entrée), alors, la valeur indiquée par l'étiquette sera l'altitude de la station la plus proche.
- Si dans les données topographiques, les coordonnées de l'entrée sont données, mais que nous voulons le dénivelé par rapport à l'entrée, alors, nous pouvons le forcer en rajoutant l'option `-value [-<altitude_de_lentree> m]`

### ***Le point Hauteur d'un décrochement (height)***

Nous pouvons afficher la hauteur d'un puits ou d'une escalade avec ce point associé à l'option :

```
-value [>hauteur> m]
```

Si nous voulons afficher un puits de 23 m, par exemple, nous écrivons (noter l'unité) :

```
-value [-23 m]
```

Si nous voulons afficher une escalade de 5 m, par exemple, nous écrirons (noter le signe) :

```
-value [+5 m]
```

Si nous ne sommes pas sûrs que cette escalade fasse bien 5 m, nous pouvons l'indiquer ainsi :

```
-value [-5? m]
```

C'est ce qui permet d'afficher par exemple P23, E5 ou E5?

### ***Le point Hauteur de la galerie (passage-height)***

Ce point fonctionne de la même manière que le point hauteur d'une marche. Il y a cependant quelques subtilités.

```
-value [5 m]
```

signifiera qu'en ce point la galerie fait 5 m de haut en bas.

```
-value [+3 m]
```

signifiera qu'en ce point le plafond est à 3 m. S'il y a de l'eau, cela veut dire qu'il y a 3 m de libre au-dessus de l'eau.

```
-value [-2 m]
```

signifiera qu'en ce point, le sol est à 2 m en dessous. S'il y a de l'eau, alors, nous comprendrons que le bassin est profond, soit 2 m de profondeur !

### ***Le point date***

Ce point permet d'afficher une date au niveau du point. Il faut lui donner la date, en écrivant dans les options (années.mois.jour) :

```
-value [2020.04.14]
```

### ***Le point Continuation (continuation)***

C'est le point que nous utilisons pour matérialiser un point d'interrogation sur la topographie. Nous pourrions en rester là, mais, bien géré, ce point permet la mise en place d'une vraie base de données de point d'interrogation, avec commentaires, que nous pouvons exporter comme une liste. Pour les fins d'exploration, ceci peut être très utile !

Pour cela, nous devons encore une fois utiliser la case option.

### ***Associer la longueur explorée, mais non topographiée***

Pour associer la longueur explorée à point d'interrogation, nous utiliserons l'option (par exemple ici 3 m) :

```
-explored 03m
```

ou (si 150 m) :

```
-explored 150m
```

Veuillez noter que :

1. il y a une unité (obligatoire !)
2. il n'y a pas d'espace entre la longueur et l'unité
3. dans le cas d'une longueur explorée inférieure à 10 m, il faut mettre un 0 pour le chiffre des dizaines. Pourquoi ? C'est un mystère...

Cette information apparaîtra dans le tableau html de continuations exporté par *Therion* (si demandé).

### **Associer un commentaire au point continuation**

Nous le faisons avec l'option :

```
-text "mon commentaire"
```

Cela fonctionne comme pour le point libellé (voir plus bas). Ce commentaire apparaîtra dans le tableau html de continuations exporté par *Therion* (si demandé).

### **Associer d'autres arguments définis par l'auteur**

Nous pouvons définir d'autres arguments à afficher dans le tableau de continuation si nous l'exportons. Pour cela, il faut utiliser la commande :

```
-attr <nom_colonne> <valeur_dans_cette_colonne>
```

Par exemple, si nous voulons une nouvelle colonne dans le tableau qui nous indique s'il y a un courant d'air, et que nous voulons appeler cette colonne avec le titre CA, alors, pour tous les points d'interrogation, nous pouvons remplir cette colonne avec :

```
-attr CA "Ca souffle"
```

Evidemment pour chaque point d'interrogation, nous pouvons changer la chaîne de caractères « Ca souffle » pour l'adapter au point, et le nouveau texte apparaîtra dans la colonne CA dans le tableau des continuations exporté.

Si cela aide à retenir le fonctionnement de cette option, Attr est l'abréviation d'attribut.

### **Le point Libellé (label)**

C'est le point à utiliser pour ajouter du texte à la topographie. La difficulté pour utiliser ce point est d'arriver à bien formater le texte que nous voulons afficher. Ceci fonctionne de façon très similaire au langage html.

Tout d'abord, avec ce point, pour écrire du texte, il nous faut utiliser l'option :

```
-text "mon texte à afficher"
```

Dans ce cas là, le texte affiché sur la topographie sera :

```
mon texte à afficher
```

Noter l'utilisation des guillemets « » qui encadre le texte à afficher. Si nous ne mettons pas de guillemets, alors, *Therion* comprend que :

- le texte à afficher est uniquement « mon »,
- que le mot « texte » est le nom d'une nouvelle option,
- que le mot « a » est l'argument de la nouvelle option « texte »
- et que « afficher » est une autre option, ici sans arguments.

Evidemment, à la compilation, ça plante, et l'erreur renvoyée est que *Therion* ne connaît pas d'option « texte » pour ce type de point. D'où l'importance des guillemets pour bien délimiter les chaînes de caractères.

Si nous voulons afficher un texte en gras, en italique, si nous voulons center le texte, si nous voulons afficher sur plusieurs lignes... il va falloir faire comme en html, utiliser des balises. Par exemple, si nous écrivons l'option :

```
-text "<center> Je centre et<br>je saute une ligne.<br><bf>cette  
troisième ligne est en gras"
```

ici, il y a trois types de balises différentes :

- `<center>` : c'est la balise qui centre le texte
- `<br>` : c'est la balise qui permet de passer à la ligne suivante (il y en a deux dans l'exemple ci-dessus)
- `<bf>` : c'est pour écrire le texte qui suit la balise en gras.

D'autres balises peuvent être utiles, comme :

- `<it>` : écrit le texte qui suit en italique
- `<ss>` : revient à du texte normal (ni gras, ni italique)

### **Le point Section (section)**

Ce point est le point qui va permettre d'afficher la section d'une galerie. Nous allons le voir plus loin, chaque section est dessinée sur dans un scrap (par exemple, 3 sections = 3 scraps différent) de projection « Aucune (None) ».

Il faut donc indiquer à ce point le scrap à afficher, qui correspond à la section que nous voulons afficher, avec l'option :

```
-scrap >nom_du_scrap>
```

par exemple :

```
-scrap SS-galerie1
```

ou SS-galerie1 est le nom du scrap de la section de la galerie1.



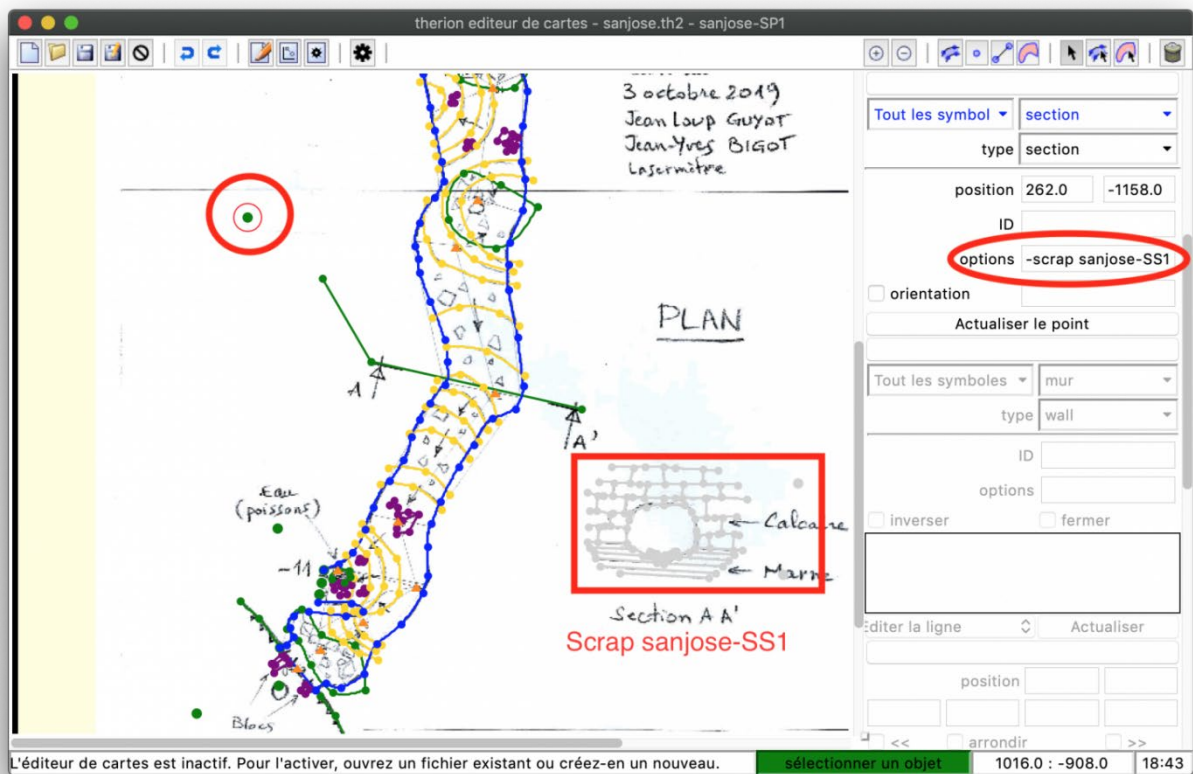
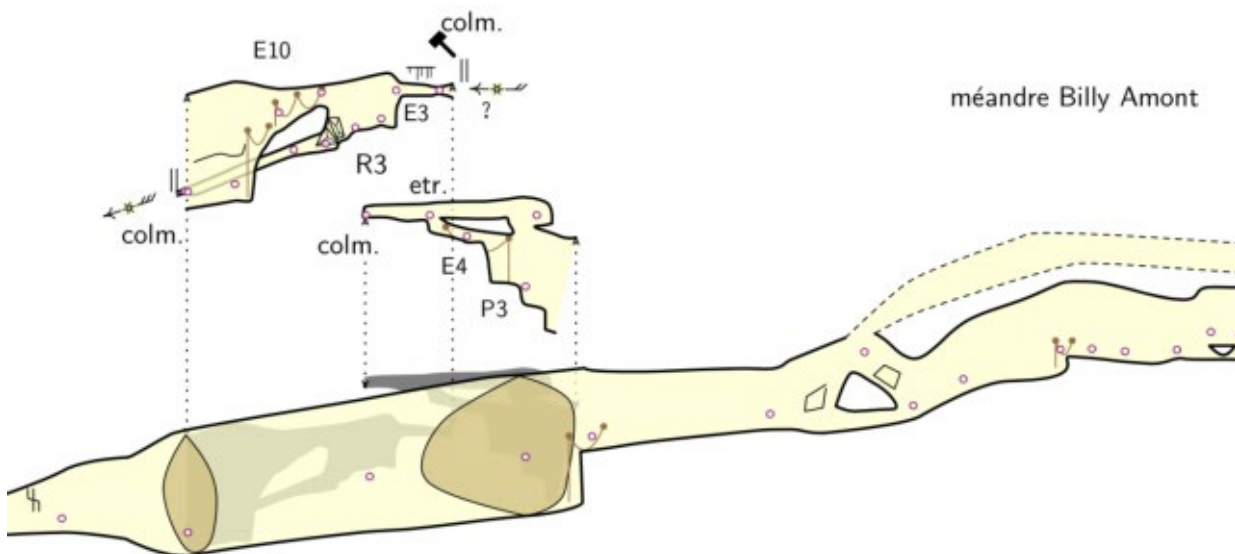


Figure 9 : Définition d'un point Section. Noter l'option qui appelle le scrap de la Section

### Le point connexion de maps (map-connection)

Ce point est complètement invisible. Il est utile lorsque nous utilisons la définition des maps (voir plus loin), et que pour des raisons de visibilité, nous décalons certaines parties du dessin (voir comment nous faisons dans la partie « notions de maps »).

Dans ce cas là, si ce point est ajouté dans la map qui sera décalée, alors, une ligne en pointillée marquant le décalage sera affichée.



Exemple de décalage (Réseau de la Combe au Puaires, Samoëns, France), avec lignes de jonction créées par les points « map-connection » dessinés dans les scraps appartenant aux maps déplacées.

Utiliser des lignes autres que la ligne mur/wall fonctionne comme cette dernière. Il suffit de changer le type. Nous pouvons aussi modifier les sous-types avec un clic-droit sur la ligne.

Sur cette page, nous ne décrivons pas toutes les lignes, mais uniquement quelques lignes un peu spécifiques.

### ***Lignes non symétriques***

Pour les lignes non symétriques, pour savoir de quel côté sera dessiné les symboles, il faut se référer au petit segment jaune au premier point de la ligne : les symboles seront dessinés du côté du segment jaune.

### ***Ligne de bord de bloc (rock-border)***

L'intérêt d'une ligne de bord de bloc, c'est qu'elle permet de définir un intérieur et un extérieur de bloc. C'est intéressant à utiliser notamment lorsque le bloc dessiné est au dessus d'une aire, comme par exemple de l'eau. A ce moment, l'eau se verra en transparence sous le bloc.

Pour cela, il faut respecter deux règles simples :

- la ligne de bord de bloc doit être fermée (le premier point correspond exactement au dernier point de la ligne)
- le petit segment jaune marque l'intérieur du bloc, et doit donc être dirigé vers l'intérieur du bloc.

Pour des dessiner des arrêtes à l'intérieur du bloc, utiliser la ligne « arrêtes de bloc » .

### ***Ligne de libellé (label)***

Cette ligne est une ligne invisible sur le dessin compilé, sauf si vous ajoutez dans la case option :

```
-text "Mon texte à afficher sur la ligne"
```

Vous pouvez formater ce texte exactement comme avec le point libellé (label). Il prend les mêmes balises.

Nous verrons aussi que cette ligne peut-être utile pour créer des aires à bord invisibles.

### ***Ligne de pente (Slope)***

Attention, dans la traduction française, deux lignes ont le même nom (Pente). En fait, elles correspondent chacune à une ligne différente en anglais :

- gradient : cette ligne va dessiner une flèche de gradient dans le sens de la pente,
- slope : cette ligne va dessiner un figuré avec des barbules, les barbules étant dirigées dans le sens de la pente.

Pour cette dernière ligne (Slope), il faut obligatoirement définir un argument supplémentaire. Si nous dessinons la ligne uniquement, sans rien faire d'autre, alors la compilation va générer une erreur.

En fait, il faut que pour l'un des points de la ligne, nous définissions la valeur de la case « dimension gauche » (L-size). Plus ce nombre sera grand, plus les barbules seront grandes. Généralement, une valeur entre 3 et 10/12 va bien.

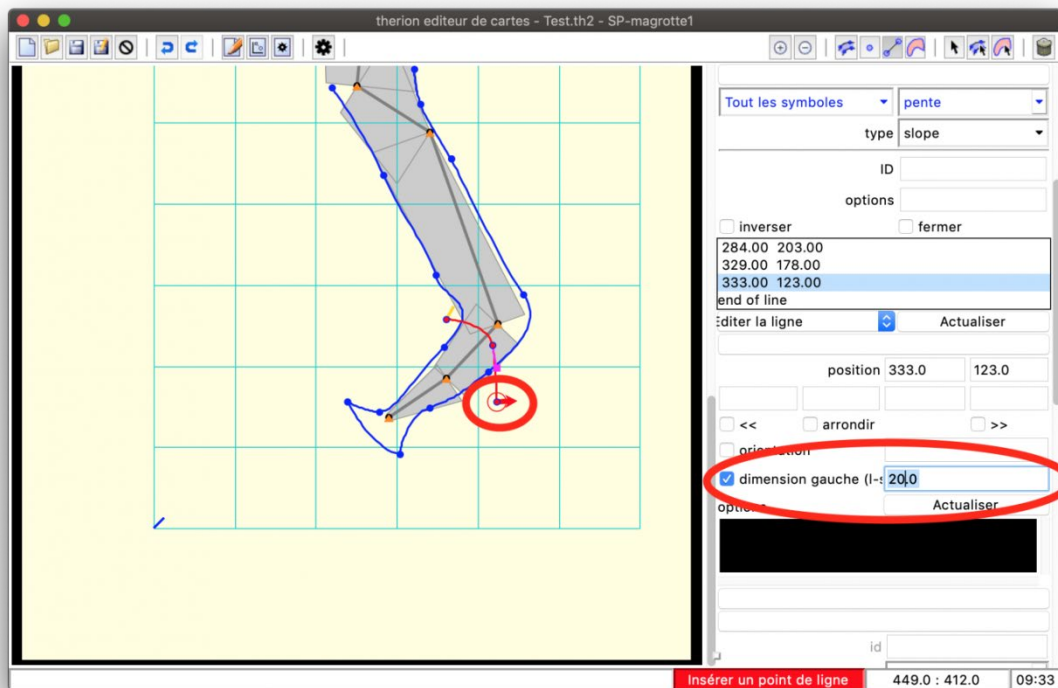


Figure 10 : Dessin d'une ligne de pente (slope).

Ici, le dernier point de ligne (sélectionné) a été utilisé pour déclarer l'option « dimension gauche », qui est imposée à la valeur 20.0. Cliquer sur le bouton « actualiser » fait apparaître sur le dessin la petite flèche rouge au niveau du point. La taille de cette flèche correspond à la longueur de la barbule sur le dessin exporté.

Attention aussi avec cette ligne, si elle est fermée et que le cercle est tout petit, nous pouvons avoir une erreur à la compilation parce qu'elle n'est pas assez longue pour calculer l'espacement des barbules.

Un conseil, quand cette ligne est utilisée, c'est de compiler à chaque fois qu'une ligne est ajoutée, afin de vérifier qu'il n'y a pas de problème. Si nous le faisons plus tard, la séance de débogage risque d'être longue et fastidieuse...

Les aires, cela va être tout les remplissages que nous voulons faire apparaître sur le dessin, comme par exemple :

- de l'eau,
- un siphon
- du sable,
- de l'argile
- des cailloux,...

En fait, nous ne dessinons pas d'aires en tant que telles. Nous ne dessinons que des lignes qui se coupent qui vont fermer une région, et nous allons déclarer ces lignes comme étant le bord de l'aire que nous voulons dessiner.

Pour dessiner une aire, il va falloir cliquer sur le bouton du bandeau supérieur « Nouvelle aire ».

### **Remarque importante**

Attention, autant pour les scraps et les lignes, ça n'a pas d'importance pour la compilation, autant, pour les aires, c'est important, très important : si vous avec la tremblote et que vous cliquez plusieurs fois sur le bouton « nouvelle aire », alors, vous aller créer plusieurs aires, mais, sauf la

dernière, elles seront toutes vides. Et ça, *Therion* n'aime pas, si l'aire est vide (c'est à dire que nous n'avons pas défini sa bordure), alors ça plante à la compilation, avec une erreur qui renvoi vers un problème d'aire.

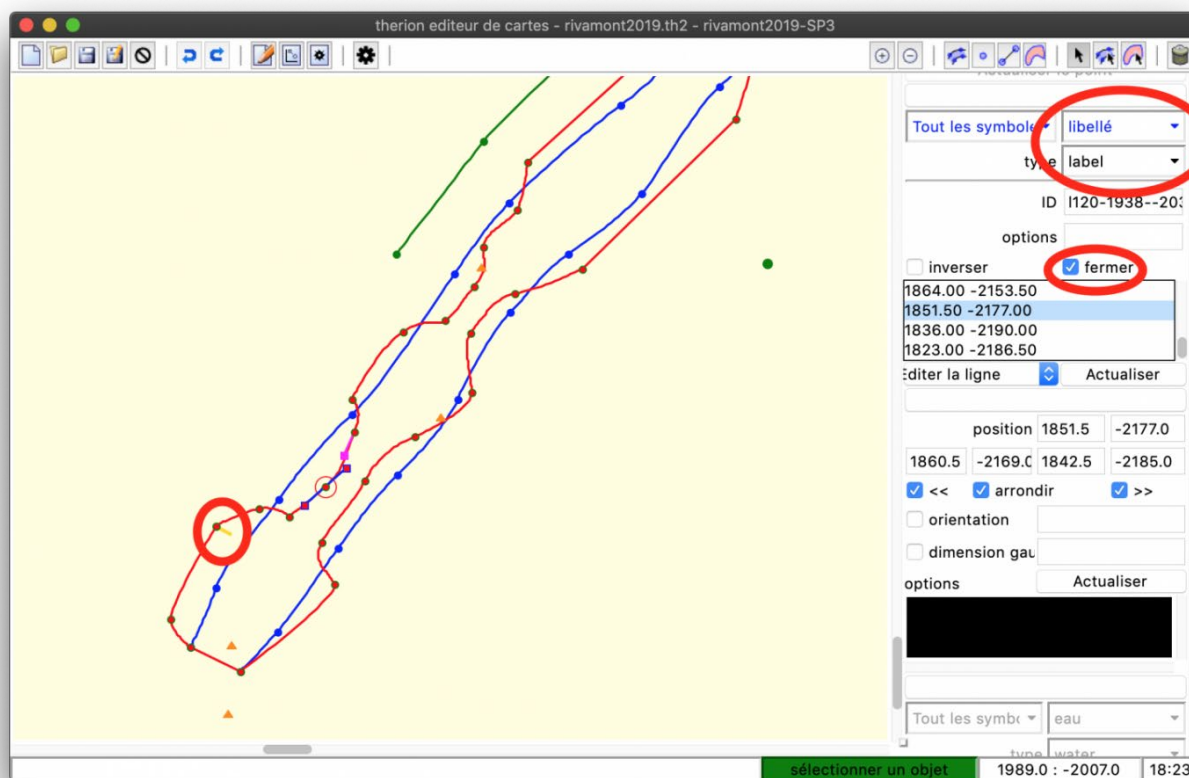
### Construire les bordures d'une aire

Pour définir les bordures d'une aire, il faut dessiner les lignes qui vont la former. Ces lignes doivent se recouper, et le petit segment jaune doit être vers l'intérieur de l'aire.

Il est possible de définir cette bordure avec plusieurs lignes, mais le plus simple, c'est plutôt de dessiner une unique ligne transparente fermée, avec le petit segment jaune pointant vers l'intérieur.

Pour dessiner cette ligne transparente, il y a plusieurs solutions :

- vous utilisez la ligne « libellé (label) » sans rajouter l'option -text « blabla ». Le texte est vide, la ligne sera invisible.
- vous utilisez la ligne « bord (border) » à laquelle vous rajouter l'option « -visibility on », soit directement en l'écrivant dans la case option, soit par un clic-droit sur la ligne



Définition d'une bordure d'aire à l'aide de la ligne « libellé (label) » invisible. Comme c'est la ligne active sur le dessin, c'est la ligne rouge. Les lignes bleues correspondent aux lignes de parois (mer (wall)). Noter le sens du petit segment jaune, ainsi que l'identifiant de la ligne (attribué automatiquement lors de la sélection pour construire une aire). Noter aussi que la ligne est fermée

### Définir les bordures d'une aire

Pour cela, cliquons sur le bouton « Nouvelle aire ». Il ne se passe rien, hormis dans le panneau de contrôle à droite qui affiche l'onglet aire.

Il va maintenant falloir définir quelles sont les lignes qui vont constituer l'aire. Nous devons cliquer sur les lignes, dans l'ordre horaire ou antihoraire (n'importe, mais il faut que les lignes se suivent).

A chaque fois que nous sélectionnons une ligne, son <id> s'affiche dans le panneau de contrôle de l'aire que nous sommes en train de construire. Attention, une fois que l'aire sera définie, si vous modifiez (coupez, rallongez,...) plus tard l'une des lignes, alors l'<id> de cette ligne sera modifiée, et elle ne sera plus reconnue dans la définition de l'aire.

C'est ici que nous nous rendons compte que si nous définissons une aire avec une seule ligne spécifique à l'aire, invisible, fermée, et bien c'est plus facile !



Figure 11 : Sélection d'une ligne fermée « libellé (label) » pour créer une aire de type eau (water)

Une fois toutes les lignes choisies, faites attention, parce qu'à chaque fois que vous cliquerez sur une ligne pendant que vous êtes dans le mode de définition de l'aire, cette ligne sera ajoutée. Il est donc conseillé, une fois que toutes les lignes ont été définies, de cliquer immédiatement sur le bouton « Sélection » du bandeau supérieur (la flèche) pour sortir du mode sélection.

### **Définir le type de l'aire**

Pour définir le type d'aire, il suffit d'utiliser l'ascenseur de gestion du type dans l'onglet Aire du panneau de configuration. Cela fonctionne exactement comme les points et les lignes.

### **Sélectionner une aire**

Comme nous le voyons, par construction, une aire n'est pas un dessin, mais est fabriquée à partir de lignes. Nous ne pouvons donc pas sélectionner une aire avec l'outil « Sélection » (la flèche), comme pour la sélection des Scraps.

Pour sélectionner une aire, il faut cliquer sur le bouton « Sélection d'aire » dans le bandeau supérieur. Nous allons ainsi sélectionner alternativement chaque aire du dessin. Il faut cliquer sur

ce bouton jusqu'à ce que les lignes de l'aire que nous voulons sélectionnée changent de couleur (deviennent rouge).

Vous pourrez alors effacer cette aire (avec le bouton « poubelle » du bandeau supérieur), ou modifier ses paramètres.

**Nous l'avons déjà vu dans la partie « Faire un dessin avec XTherion ». Mais c'est un point important, nous reprenons donc ce qui a déjà été écrit :**

Une règle de base pour dessiner avec XTherion, c'est que nous sommes obligés de dessiner dans des calques, que nous appelons **Scraps** dans *Therion*.

Ceux qui sont habitués à utiliser Visual Topo et un logiciel de DAO sont habitués au système de calques. Mais, chaque personne a sa propre méthodologie ; certains utilisent un calque par type d'objet dessiné (fond, parois, puits, texte,...), d'autres par zone dessinée (avec pourquoi pas des sous-calques par type d'objet) et même certains, mettent finalement tout dans un même calque sans se soucier d'une quelconque organisation.

Sous *Therion*, le fonctionnement des Scraps est différent et demande de la rigueur et de la logique. En fait, chaque scrap correspondra à un bout de topographie dessiné complètement (il y a tout, les murs, les stations, le texte,... tous les symboles, en fait !).

Vous allez dire que nous pourrions finalement dessiner toute la topographie sur un seul scrap, ce serait plus simple. Dans certains cas, ce serait possible :

- **si aucune galerie ne se superpose :**
  - nous allons le voir, s'il y a superposition de galerie, le dessin de chaque bout sur des scraps différents permet à *Therion* de gérer la transparence automatiquement : sur le pdf final, nous verrons la galerie inférieure en transparence ! Et ce, sans s'embêter à le faire manuellement !
  - nous le verrons aussi plus tard, mais les parois des galeries ne doivent pas se croiser !
- **et si la topographie n'est pas trop grande :** en effet, un scrap énorme sera difficile à exploiter car la réaction de l'éditeur de dessin sera lente voir très lente...

En fait, ces deux points sont finalement deux points qu'il faut avoir en tête lorsque nous prévoyons de créer un nouveau scrap :

- Il **doit être petit** (idéalement contenir moins d'une trentaine de stations topographiques) ; lors de la compilation, plus les scraps sont grands, plus le risque d'avoir un dessin déformé sur une grande zone est important
- il **ne doit pas comporter de galeries se superposant** : s'il y a des endroits qui se superposent il faut un scrap par niveau de superposition
- et le corollaire du point précédent, pour permettre la gestion automatique de l'ordre de superposition des scraps et de la transparence sur le dessin final, il est bon d'**avoir une idée des altitudes relatives des stations topographiques** dans le scrap : par exemple, supposons deux galeries pentues qui se superposent. Il faut que toutes les stations contenues dans le scrap supérieur soient plus hautes que toutes les stations contenues dans le scrap inférieur. Vous le comprendrez rapidement par la pratique, et c'est finalement le point le plus difficile à mettre en œuvre pour faire un beau dessin. Voir la figure ci-dessous :

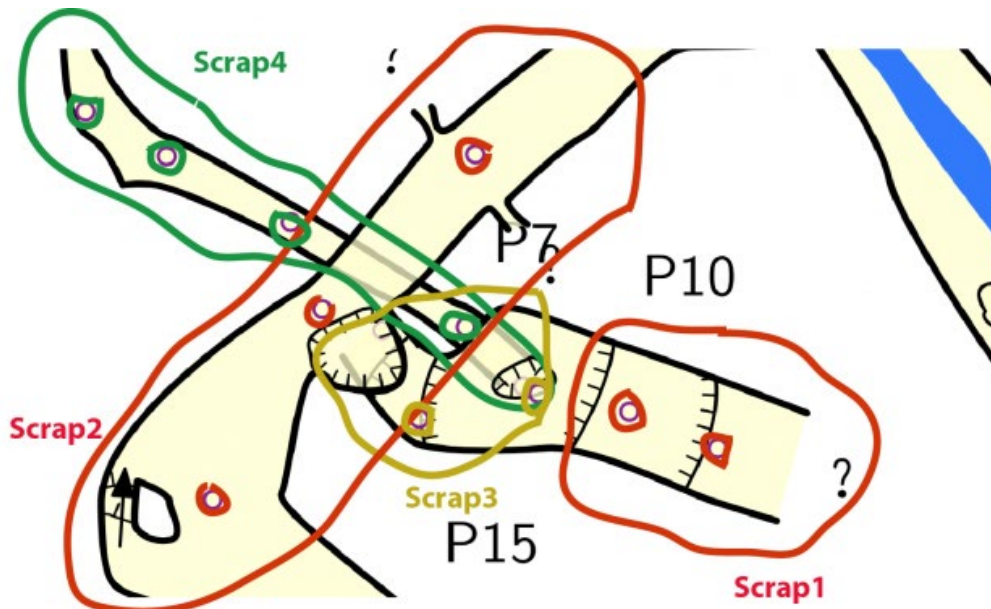


Figure 12 : l'exemple d'une zone complexe avec différents niveaux de superpositions

Ici, idéalement, il faut découper en 4 scraps différents.

Enfin, *Therion* doit connaître le rapport d'échelle du scrap. Ceci peut se faire de deux façon, soit en le définissant graphiquement et en utilisant le panneau de contrôle dédié, soit grâce aux stations topographiques utilisées dans le scrap. Dans ce cas, **il faut a minima que le scrap contienne 2 stations topographiques.**

Ce point peut être intéressant à comprendre pour améliorer le lever topographique sous terre : si nous voulons matérialiser correctement un départ dans une galerie ou un puits, où il y a une superposition, alors, il faut qu'il y ait a minima 2 stations topographiques matérialisées dans ce départ... Ce qui souvent manque et peut poser des difficultés (surmontables, mais de façon brouillonne) lors du dessin.

Dans le cas d'un travail sur une cavité complexe ou sur un système karstique complet, au fur et à mesure de nos explorations, nous allons devoir ajouter de nouvelles topographies.

Nous avons déjà vu comment joindre des données topographiques avec un fichier maître. Pour le dessin, c'est pareil, il va falloir les joindre, sans quoi sur l'export final, les zones entre chaque dessin ne seront pas jointes de façon esthétique.

Prenons l'exemple de deux scraps (SP-scrap1 et SP-scrap2 par exemple) à joindre.

- Si ces deux scraps sont dessinés chacun dans la continuité de l'autre dans le même fichier th2, alors, il suffit de faire correspondre les lignes de mur/wall (en pratique, il faut que les points de début/fin de ligne se superposent entre les deux scraps).
- Si ces scraps sont dans deux fichiers différents, ou dans le même fichier de dessin, mais qu'ils ne sont pas dessinés dans la continuité l'un de l'autre, alors, dans le fichier maître, il faudra :
  - importer les fichiers de dessin,
  - rajouter la commande
    - `join SP-scrap1 SP-scrap2 # si jointure à un seul endroit`
    - `join SP-scrap1 SP-scrap2 -count 2 # si jointure en deux endroits`

Dans ce cas, le fichier maître topographie\_globale.th deviendrait :

```
# déclare la nouvelle survey qui va être la somme des deux surveys
galerie1 et galerie2 ;
# nous l'appelons MaGrotte
```

```

survey MaGrotte
# Appel des fichiers de données
input bout-trou1.th
input bout-trou2.th
# impose la première jonction
equate 3@galerie1 3@galerie2
# impose la seconde jonction
equate 10@galerie1 argile27@galerie2

# Appel des dessins
input bout-trou1.th2
input bout-trou2.th2

# Fait les jonctions de dessin
join SP-scrap1 SP-scrap2 -count 2
#fin de la survey
endsurvey

```

Ici, nous laissons *Therion* décider au mieux ce qu'il faut joindre. Nous pourrions vouloir imposer la jonction de telle ou telle ligne. Pour cela, il nous faut avoir donné un identifiant (nom) pour chaque lignes que nous voulons joindre.

Nous pouvons déclarer un nom pour une ligne en sélectionnant cette ligne, et en donnant le nom dans la case <id> du panneau de configuration (onglet ligne).

Ensuite, si nous voulons joindre deux lignes :

- la fin d'une ligne nommée « ligne1 » du scrap « SP-scrap1 »
- avec le début de la ligne nommée « ligneX » du scrap « SP-scrap2 »

nous écrivons :

```
join ligne1@SP-scrap1:end ligneX@SP-scrap2:0
```

Il peut être utile de rajouter le dessin d'une section de galerie sur un plan ou une coupe. Nous ne pouvons pas le faire directement dans le scrap de plan ou coupe dans lequel nous voulons dessiner la section. Il faut pour cela créer un scrap spécifique pour chaque section que nous voulons dessiner. Chaque section doit être dans un scrap différent.

### ***La projection d'un scrap de section***

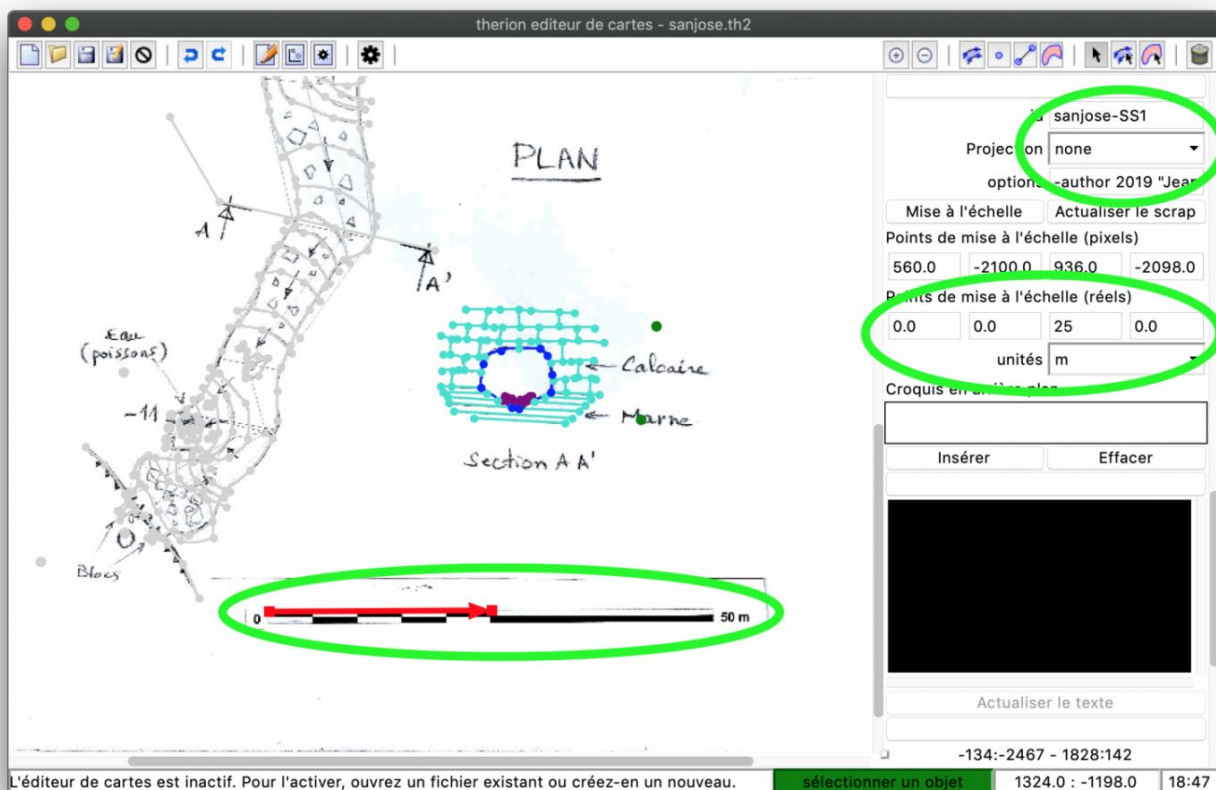
Chaque scrap de section doit être défini avec la projection « Aucune (None) ».

### ***Comment définir l'échelle du Scrap de section ?***

Dans la partie du tutoriel expliquant comme nous construisons des scraps, nous avons expliqué que chaque scrap doit pouvoir être mis à l'échelle. Pour les dessins des plans et coupes, nous utilisons les stations topographiques. Pour un scrap de section, ce n'est pas possible généralement. En effet, le dessin d'une section ne comporte généralement pas de station topographique, où à la rigueur, une seule station topographique.

Mais lors de la création du nouveau scrap de section, sur le plan de dessin, en bas à droite, une flèche rouge apparaît. Il suffit de tirer les extrémités de cette flèche pour les adapter à une distance connue. Dans le panneau de contrôle du scrap (à droite), ça met à jour les 4 cases du contrôle de l'échelle de l'image (« Point d'échelle de l'image (picture scale points) »).





Définition d'un scrap pour dessiner une section de galerie. Noter le nom, du scrap et sa projection. Cette figure montre aussi comment définir l'échelle du scrap : nous plaçons la flèche rouge sur l'échelle, entre 0 et 25 m (c'est horizontal), et nous modifions ces coordonnées dans les points de mise à l'échelle (panneau de contrôle à droite)

Toujours dans ce dans ce panneau de contrôle à droite, dans l'onglet scrap, juste en dessous il suffit alors de renseigner les valeurs de ces points pour en vrai. Par exemple, si la flèche est horizontale et si la longueur de la flèche correspond à 10 m sur le terrain, alors, nous écrivons dans les 4 cases :

0.0 0.0 50.0 0.0

les cases signifient :

"coordonnée X point départ"    "coordonnée Y point départ"    "coordonnée X point arrivée"    "coordonnée Y point arrivée"

C'est là où nous voyons l'utilité d'avoir une grille de taille connue en arrière plan, ou d'avoir un repère d'échelle sur l'image de fond que nous utilisons.

Voilà, c'est mis à l'échelle. Il suffit maintenant de dessiner comme pour un scrap normal. Sur la topographie finale, la section sera mise à l'échelle de la topographie.

### **Comment intégrer le scrap Section dans le plan ou la coupe ?**

Nous l'avons vu dans la partie décrivant quelques points spécifiques, avec le point « Section » que nous utilisons dans le scrap de plan ou de coupe où nous voulons insérer la section. Voir cette description pour plus d'information.

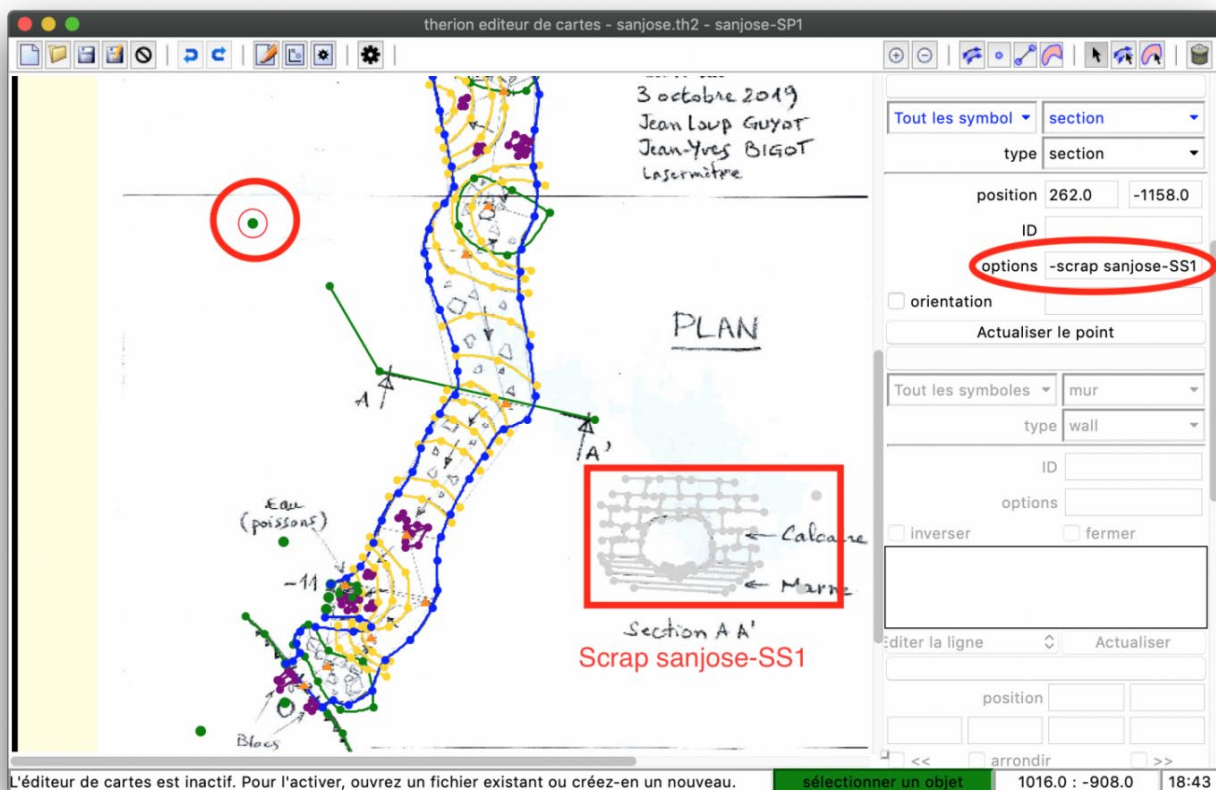


Figure 13 : Définition d'un point Section. Noter l'option qui appelle le scrap de la Section

Au niveau de la méthodologie de dessin, dessiner une coupe développée ne diffère pas de la technique utilisée pour dessiner un plan. La différence entre les deux ne sera que dans la définition de la projection du scrap qui sera alors « extended »

Pour les coupes développées, avant de dessiner quoi que ce soit, il va falloir préparer le squelette topographique en conséquence. C'est comme sur Visual Topo, il va falloir décider ce que nous étendons vers la droite ou vers la gauche. Pour les cavités complexe, c'est la partie la plus difficile parce qu'elle demande de bien visualiser ce que peut donner telle ou telle définition d'extension. La difficulté pour travailler les coupes développées n'est pas dépendante du logiciel utilisé, mais uniquement de nos propres capacités à concevoir le rendu que nous voulons.

Il faut aussi bien comprendre qu'avec *Therion*, contrairement à ce que nous pouvons faire avec Visual Topo, nous ne pouvons pas couper un bouclage dans le calcul. Dans ce cas, *Therion* coupe automatiquement la boucle en considérant que la galerie principale est la plus courte. Si nous ne sommes pas contents du résultat donné automatiquement par *Therion*, il va donc falloir trouver où casser les bouclages de façon à ce que le résultat soit conforme à ce que nous voulons obtenir.

Ces deux points se traitent en amont du dessin, directement dans les fichiers de données topographiques qu'il va nous falloir modifier, grâce à la commande :

```
extend <left, right, normal, reverse, vertical, start, hide, ignore, [0-200]> [<station_1> [<station_2>]]
```

Dans certains post à propos de *Therion* sur les forums, il est parfois mentionné (à tort) :

mais cette commande renvoi l'erreur :

```
unknown extend flag -- break
```

### **Gestion du côté d'extension d'une coupe développée**

Par défaut, une série de données topographique est étendue de la gauche vers la droite (sens de lecture normal). Nous pouvons modifier ce comportement en éditant le tableau dans une centerline donnée d'un fichier de données topographiques.

### **Modifier le sens d'extension d'un bloc de visées topographiques**

Supposons que les visées entre les stations 1 et 5 sont vers la droite (la valeur par défaut), mais que nous voulons étendre les visées entre les stations 5 et 8 vers la gauche, et que nous voulons que la suite soit vers la droite. Nous modifierons le tableau de données (dans une centerline, elle-même dans une survey) ainsi :

```
data normal from to ...
1 2 ...
2 3 ...
3 4 ...
4 5 ...
extend left
5 6 ...
6 7 ...
7 8 ...
extend right
8 9 ...
9...
```

La commande « extend right/left » va s'appliquer pour toutes les visées qui la suivent. Pour une cavité simple, c'est facile à gérer.

Nous pourrions écrire un fichier de données (.th) spécifique, dans lequel, nous ne ferions que déclarer les valeurs d'extension (dans un bloc centerline-endcenterline). Dans le cas de l'exemple ci-dessus, nous écririons :

```
centerline
  # si besoin de rajouter des indications de survey
  station-names []@ma_survey
  extend right 1 5
  extend left 5 8
  extend right 8 ...
endcenterline
```

Pour une cavité complexe, c'est beaucoup plus difficile, il faut le faire au fur et à mesure, de façon très rigoureuse. Il peut-être bon d'exporter le pdf d'un plan avec le nom des stations et la ligne de centerline, avec ces options dans le *thconfig* :

```
#symbole-hide line survey # commenter cette ligne
debug station-names
```

Ensuite à chaque insertion d'une commande « extend », compiler pour vérifier que le résultat correspondent bien à ce que vous voulez.

### **Modifier le sens d'extension d'une visée topographique**

Pour modifier le sens d'extension d'une visée spécifique, nous pouvons le faire

- en encadrant la visée avec une commande centerline right/left avant et une autre, dans le sens contraire, après.

- ou plus simplement en indiquant la visée à modifier à la suite du right/left ; Par exemple, pour imposer l'extension vers la gauche uniquement de la visée de la station 5 à la station 6 :

```
extend left 5 6
```

Nous pouvons aussi imposer à une visée d'être représentée verticale. Par exemple, pour imposer la visualisation de la visée 5 à 6 verticale, nous écrivons :

```
extend vertical 5 6
```

Nous pouvons aussi remplacer les paramètres right ou left par normal ou reverse (= inverse en anglais). Cela veut dire que la direction d'extension des visées suivantes sera normale ou inverse par rapport à la visée précédente.

Enfin, avec les versions de *Therion* >5.5.1, il est possible de rajouter un coefficient (entre 0 et 200) à la commande. Ce coefficient permet de raccourcir l'extension d'une visée. Par exemple :

```
extend 100 1 2
```

est équivalent à :

```
extend right 1 2
```

Ou :

```
extend 0 1 2
```

est équivalent à :

```
extend vertical 1 2
```

Ou :

```
extend 200 1 2
```

est équivalent à :

```
extend left 1 2
```

### **Casser artificiellement un bouclage ?**

Si nous avons une boucle topographique, dans le cas d'un export d'une coupe développée du réseau, *Therion* va être obligé de couper la boucle. Il n'y a pas le choix. Si rien n'est spécifié, il coupe la boucle selon sa propre logique, en considérant la partie la plus courte comme le cheminement principal, et sur la coupe développée exportée sans l'import de dessin, nous verrons une double flèche pointillée qui indiquera la zone de coupure et la jonction.

Cette coupure peut ne pas vous convenir. Il faut alors imposer une coupure où nous la voulons. Ce n'est pas toujours facile à faire, il faut souvent procéder en essai-erreur sur les réseaux complexes.

Pour cela, nous devons indiquer quelle visée nous voulons couper. Si par exemple, nous voulons couper au niveau de la visée des stations 4 à 5, nous écrivons dans le tableau de données :

```
extend ignore 4 5
```

Il se peut que cela ne fonctionne pas comme vous le vouliez. En ce cas, tentez :

extend ignore 5 4

En effet, l'argument « ignore » est lui même orienté.

Pour couper à la station 4, pour les versions de *Therion* >5.4.4, il est possible d'utiliser la commande :

extend ignore 4

Attention :

- Si vous utiliser une version 5.4.4 ou inférieure, il y a un bug, et en fait, cette dernière ligne va couper non pas à la station 4, mais à la station 3. Ce bug est corrigé pour les versions suivantes. Dans ce cas là, soit il faut éviter d'utiliser la commande avec une seule station (indiquer une visée à la place), soit il faut mettre à jour le logiciel.
- Si nous voulons couper au niveau d'un des carrefours formant la boucle, alors, si nous utilisons cette dernière commande avec l'indication d'une seule station topographique, alors, *Therion* ne sait pas quelle visée couper (il a le choix entre 2 a minima), il va en prendre une qui ne sera pas forcément celle que vous voulez. Donner la visée au lieu de la station en argument est alors plus judicieux.

Normalement, avec ces quelques définitions et exemples, vous devriez être armés pour concevoir puis dessiner de belles coupes développées. Sinon, vous pouvez vous référer à la page <https://Therion.speleo.sk/wiki/breakingextend> du wiki (en anglais) qui traite de ces problèmes de manière extensive.

Cette page est une fiche synthétique récapitulative des différentes étapes que nous devons suivre pour faire un dessin avec *XTherion*. Pour ceux qui lisent l'anglais, ne pas hésiter à se référer aussi à l'[excellente page du wiki Therion](#).

1. Produire un fichier de base, soit un fichier xvi, soit une image au format .gif
  1. Pour produire un fichier xvi, utiliser la commande export dans le *thconfig*, avec uniquement l'import des données topographiques actif
  2. Compiler
  3. Décommenter (ou écrire) l'import du dessin que nous allons faire
2. Dans l'éditeur de dessin de *XTherion*
  1. Ouvrir un nouveau dessin
  2. Importer le fichier de fond
    1. Enregistrer le fichier dessin (.th2)
    2. Choisir le fichier de fond à insérer
  3. Faire un nouveau Scrap
    1. Bien penser à indiquer le nom, la projection et les options si besoin
    2. Attention à bien choisir l'extension du scrap (quelles stations topographique vont appartenir à ce scrap)
  4. Insérer les points stations
    1. Voir s'il y a besoin de dire à quelle topographie ils appartiennent
  5. Compiler pour voir s'il n'y a pas d'erreur dans les stations
  6. Dessiner les lignes de parois (mur/wall)
    1. Attention au petit segment jaune pour définir l'intérieur
    2. Attention aux parois non fermées et ouverte sur l'extérieur
    3. Attention à ne pas se faire croiser les lignes de parois
    4. Dans le cas particulier des piliers, penser à rajouter l'option -outline in
  7. Compiler pour vérifier que tout est en ordre
  8. Rajouter les autres figurés sous forme de points
  9. Rajouter les autres figurés sous forme de lignes
  10. Rajouter les autres figurés sous forme d'aires
    1. Doit systématiquement être après le dessin des lignes
    2. Attention à ne pas définir d'aires vides

11. Compiler régulièrement pour éviter les erreurs et faciliter la recherche d'erreurs si besoin
12. Faire une mise à jour de la définition des maps (voir plus loin dans les pages...)
13. Faire un nouveau scrap
  1. Reprendre les étapes ci-dessus
  2. Si ce nouveau scrap n'est pas dans le même fichier dessin, alors importer le nouveau fichier dessin dans un fichier maître ou dans directement dans le *thconfig*
  3. Gérer les jonctions de scraps

## 8. DES ERREURS CLASSIQUES

La compilation peut générer des erreurs. Il est important de savoir les repérer, les comprendre et les corriger.

Au cours de la compilation, *Therion* peut générer des warnings, ou des erreurs.

Ces dernières stoppent la compilation, et nous n'avons pas de documents exportés. Les erreurs sont donc graves parce qu'elles empêchent la compilation. Les erreurs sont donc faciles à détecter, toutes les informations se trouvent dans le log de compilation. Il faut juste savoir les lire, et les comprendre.

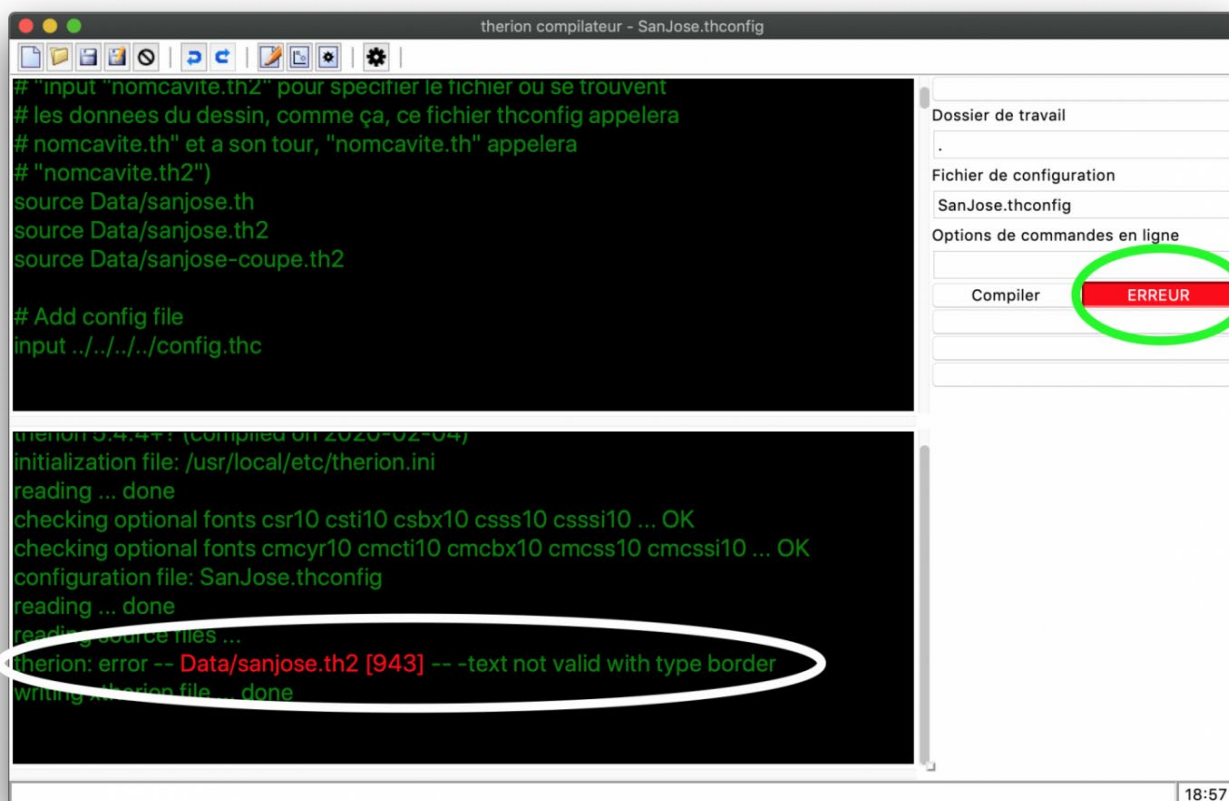


Figure 14 : Fenêtre de compilation de *XTherion* avec une erreur de compilation.

La case à droite est rouge, et l'avant dernière ligne indique qu'il y a une erreur dans le fichier de dessin *Data/Sanjose.th2*, parce qu'à la ligne 943, on utilise une ligne de type « bord (border) » avec l'option `-text`, qui n'existe pas pour ce type de ligne. Si nous supprimons cette option, alors, la compilation se fait bien

Les Warnings sont moins problématiques parce qu'ils n'arrêtent pas la compilation. Nous obtenons donc nos exports. Seulement, ils peuvent tout de même impacter le rendu final,

notamment pour l'export en 3D ou l'export de fichiers SIG tels que les kml ou les shapefiles. Ces fichiers risquent de ne pas afficher le scrap qui génère un warning.

Les Warnings sont plus difficiles à repérer. Ils sont aussi indiqués dans le log de compilation. Ce dernier dit généralement quel est le scrap posant problème, mais résoudre le problème est parfois difficile parce qu'il faut trouver l'objet qui pose problème.

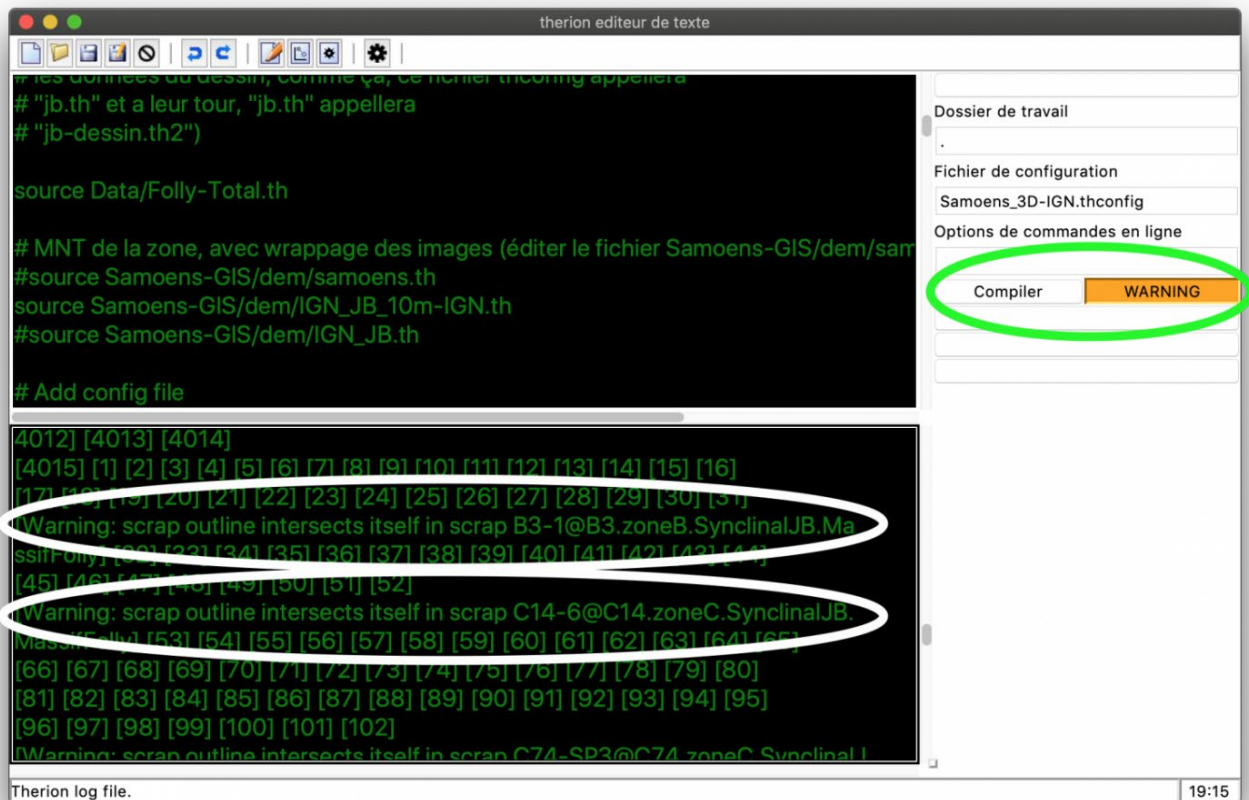


Figure 15 : Exemple de compilation avec des warnings.

La case est orange, et dans la fenêtre du bas, nous pouvons voir qu'il y a deux warnings (plus un troisième coupé en bas). Le premier Warning dit que dans le scrap B3-1 de la cavité B3 qui est dans zoneB, qui appartient à synclinal JB dans le Massif du Folly, une ligne de paroi en croise une autre, alors qu'elle ne devrait pas. Le second warning est identique, sauf qu'il est pour le scrap C14-6, de la cavité C14 de zoneC dans SynclinalJB, toujours dans MassifFolly. Hormis le nom du scrap, tous les noms donnés sont des surveys imbriquées. La survey B3 est dans la survey zoneB qui est dans la survey SynclinalJB qui est dans la survey MassifFolly.

Dans la suite nous allons voir comment comprendre ces erreurs, et comment les résoudre.

Lorsque la compilation plante avec une erreur, dans la fenêtre log (ou dans le fichier *Therion.log*), les dernières lignes décrivent l'erreur. Ces lignes donnent généralement ces informations :

- une description rapide de l'erreur
- le fichier qui a généré l'erreur
- la ligne du fichier qui a généré l'erreur

Dans ces cas là, il est important de bien lire ces informations pour pouvoir corriger l'erreur. Ça se gère au cas par cas. Cette page présente quelques cas classique, mais est loin d'être exhaustive.

## ERREURS AVEC LES FICHIERS D'ENTREES

Lors de la compilation, surtout si vous avez travaillé sur les fichiers de compilation, des fichiers maîtres ou si vous avez rajouté des dessins, des scraps, des maps..., *Therion* peut générer une erreur, vous disant qu'il ne trouve pas tel ou tel fichier, ou tel ou tel scrap/map.

### **Erreurs dans les chemins de fichiers**

Généralement, il vous donne le nom du fichier ainsi que la ligne où le fichier non trouvé est importé.

Souvent, c'est parce que le nom du fichier, ou son chemin possède une faute. Essayer de la trouver, et de la corriger. C'est souvent facile à corriger.

Nous le répétons, n'utiliser pas d'espaces, ni de caractères accentués dans les noms de fichiers, ça va poser ce type de problèmes !

### **Erreurs dans noms de survey, scrap, map**

Vous pouvez recevoir une erreur du type (avec le fichier et la ligne où se trouve l'appel) :

```
Error : survey/map/scrap MaTopo n'existe pas ou n'a pas été trouvée
```

Trois types d'erreurs peuvent générer ce type de message

1. Vous n'avez pas mis à jour tous les fichiers th, vous avez par exemple oublié d'importer le dernier fichier th2 de dessin que vous avez fait,... Il faut donc vérifier que vous avez bien importé tous les fichiers, et que vous avez bien mis à jour tous vos blocs map et survey avec votre nouveau dessins
2. Il y a une faute d'orthographe dans le nom de l'objet non trouvé : vérifiez tous les noms de ces objets pour vérifier que l'erreur ne provient pas de là.
3. L'objet n'est pas trouvé, parce que vous n'avez pas spécifié à qui il appartient, avec l'argument @masurvey ou @mamap,... Vérifier que vous ne vous êtes pas planté à ce niveau là

## ERREURS AVEC L'EXPORT

### **Erreur de mémoire**

Ce type d'erreur peut arriver lorsque nous essayons d'exporter en pdf un réseau complexe, avec de très nombreuses stations topographiques, mais sans aucun dessin. Normalement, nous devrions obtenir un pdf, avec juste les stations et la ligne de cheminement.

Sauf que s'il les données topographiques comporte un très grand nombre de stations, alors, nous recevons une erreur, avec un message complexe, plein de chiffre et la notion de mémoire (memory, en anglais) dépassée.

Pourquoi ? C'est que dans le cas de ce type d'export, ou nous n'utilisons pas de dessin, *Therion* considère toutes les données topographiques comme un seul scrap. Par construction du programme *Therion*, la taille de la mémoire allouée est limitée, et c'est ça qui génère l'erreur.

Généralement, découper ce réseau complexe en bout de topographie plus petits, et les compiler un à un séparément permet d'afficher tout de même chaque partie de réseau.

Sinon, il faut dessiner et importer les dessins.



## ***Erreur de taille de scrap***

A la compilation, avec un dessin, nous pouvons recevoir l'erreur que l'un des scraps est trop grand et qu'il y a un problème d'échelle (avec indication du fichier de dessins, et du scrap correspondant).

Cela veut dire qu'il y a un problème dans les coordonnées locales (du dessin) du scrap en question. Il faut trouver où c'est. Cela peut être du à :

- un mauvais rapport d'échelle dans la définition du scrap problématique : la vérifier
- à la présence de 2 entrées qui ont eu leurs coordonnées fixées, mais avec une entrée possédant une coordonnée fautive de beaucoup
- ou à un élément de dessin ayant eu une de ses coordonnées corrompues (par exemple lors de l'édition du fichier th2 dans un éditeur de texte). C'est plus difficile à déboguer. Une solution (fastidieuse, mais qui fonctionne), c'est :
  - éditer le fichier .th2 dans un éditeur de texte,
  - commenter toutes les lignes à l'intérieur du scrap problématique
  - Compiler.
    - Si l'erreur est encore présente, c'est que l'erreur provient de la définition du scrap
    - si non, passer à la suite
  - décommenter deux lignes de stations topographiques
  - compiler
  - décommenter une à une les lignes des points de stations topographiques, en compilant après chaque ligne dé-commentée
  - décommenter les lignes de paroi une à une, en compilant entre chaque ligne dé-commentée
  - décommenter les autres objets un à un, en compilant entre chaque objet dé-commenté, jusqu'à ce que nous trouvions l'erreur

## ***Erreur de taille d'export***

Nous obtenons ce type d'erreur généralement pour les exports en pdf, de grosses cavités à de grandes échelles. En effet, la taille maximale d'un pdf est limitée. Du coup, si l'échelle d'impression que nous avons demandée fait que notre topographie finale est plus grande que la taille maximale autorisée d'un pdf, nous obtenons cette erreur.

Pour y remédier, diminuer l'échelle. Par exemple, si nous avons ceci :

```
scale 1 500      alors, tester avec cela :  scale 1 1000
```

## **Erreurs dans les dessins**

### ***Aire avec des lignes répétées (area with repeated lines)***

Nous recevons cette erreur lorsqu'une même ligne a été définie deux fois pour cette aire là, nous avons cliqué deux fois sur la même ligne.

Pour trouver où est le problème, naviguer dans les différentes aires (bouton « sélection aire »), et regarder pour chaque aire dans le panneau s'il n'y a pas un identifiant de ligne en double. Quand vous la trouvez, sélectionner l'identifiant dans ce panneau, et effacer l'appel en double

### ***Aire Vide***

Si vous avez cette erreur ou si vous avez une erreur bizarre dans un scrap, c'est peut-être du à une aire vide.

Pour corriger, éditer le fichier de dessin correspondant à l'erreur, et rechercher toutes les instances d'aires vides qui vont être sous la forme :

```
area  
endarea
```

 puis les effacer.

Attention, pour modifier un fichier de dessin dans un éditeur de texte, il ne doit pas être ouvert dans l'éditeur de dessin de *XTherion*.

### **Élément en dehors d'un scrap**

Cela arrive lorsque vous avez dessiné des objets (points, ligne, aires) en dehors d'un scrap, souvent en ayant oublié de définir un scrap. *Therion* renvoie généralement :

```
missing scrap before line command (Manque un scrap avant la ligne de commande)
```

Essayer de trouver l'objet (ou les objets) en question, et voir s'il (ils) ne vont pas dans un scrap existant. Si c'est le cas, cliquer sur l'objet, et dans le panneau de contrôle des éléments, l'objet en question est sélectionné. Dans ce panneau de contrôle, il y a une ligne qui permet de déplacer l'objet sélectionné vers le scrap dans lequel nous le voulons [insérer image]

S'il manque un scrap, créer un nouveau scrap, puis de la même manière qu'au dessus, soit vous déplacez les objets dans le nouveau scrap, soit vous déplacez les instances begin scrap ou endscrap qui correspondent au scrap dans lequel vous voulez ces objets.

### **Erreurs d'options**

Parfois, *Therion* se plaint qu'une l'option d'une commande (et il vous dit la ligne et le fichier en question) n'existe pas (wrong option). C'est que vous vous êtes trompé dans le nom de la commande, vérifiez-la.

Aussi, *Therion* peut vous renvoyer l'erreur :

```
Wrong argument for option XXX (argument faux pour l'option XXX)
```

ou

```
Too many/ not enough argument for option XXX  
(trop/pas assez d'arguments pour l'option XXX)
```

Dans les deux cas, aller à la ligne du fichier correspondant à l'erreur, et vérifier les arguments de l'option. C'est généralement facile à déboguer.

## **ERREURS DE DÉFORMATION DU DESSIN COMPILÉ**

Vous avez compilé votre projet, et *Therion* ne vous a pas insulté. C'est bien, vous êtes content, vous avez votre sortie. Mais pas de bol, votre dessin n'est pas joli, il y a certaines zones très déformées, et ce n'est pas normal... Il faut corriger.

Il peut y avoir deux causes différentes :

- La première, c'est que vous avez plusieurs scraps, et qu'avec un commande « join », vous vous êtes trompés de scraps et vous avez imposé la jointure de 2 scraps qui ne sont pas contigus. *Therion* est bête, il vous obéit au doigt et à l'œil, alors, il joint les deux bouts ouverts les plus proches de ces deux scraps, et ce n'est pas joli à voir
- La seconde, c'est que vous avez fait une erreur dans le nom déclaré d'une station d'un des points « station topo » que vous avez dessiné. Rappelez-vous, le dessin est ancré sur le

squelette topographique, et il est déformé si le squelette est modifié par la suite. Cela veut dire, que si vous vous êtes trompé de nom dans la déclaration d'un point « station topo », alors, tous les éléments autour de ce point vont être attirés (et donc déformés) vers la vraie localisation du point topo mal nommé. Ici, vous voyez tout l'intérêt de travailler avec de petits scraps, si vous ne trouvez pas l'erreur, cela demandera peu de travail pour effacer ce scrap et le refaire, en dernier recours.

Pour remédier au premier point :

- regarder dans un premier temps ces commandes « join », pour voir si vous ne voyez pas une erreur évidente.
- si vous ne trouvez pas, commenter toutes les lignes « join »
- puis décommenter une à une ces lignes join, en compilant entre chaque ligne décommentée et en regardant le résultat.
- Corriger la ligne qui pose alors problème !

Pour remédier au second point :

- Il faut éditer le scrap dans l'éditeur de dessin, et, aux alentours de la zone déformée, vérifier que le nom de toutes les stations correspond bien à la station correspondante.

### ERREURS BIZARRES...

Il peut arriver (rarement, heureusement !) d'obtenir des erreurs bizarres, difficilement compréhensibles.

#### ***Therion affiche une erreur Tex, et tourne en boucle sans s'arrêter***

De par notre pratique, ceci nous est arrivé avec l'utilisation de la ligne de type « Corde (rope) ». Cette ligne est parfois difficile à utiliser, il ne faut pas arrondir les angles de cette ligne. Si une partie de la ligne sort de la galerie (i.e. recoupe une ligne de paroi), alors, cette erreur est générée.

Pour éviter cette erreur :

- Eviter de faire des lignes « Corde (rope) » trop complexes
- Vérifier que ces lignes ne recoupent pas une ligne de paroi (mur/wall)

Une bonne pratique, c'est, lorsque nous dessinons, de compiler très souvent. Plus nous compilerons souvent, plus nous pourrons détecter les erreurs de dessin rapidement, et ainsi contraindre plus facilement d'où elles viennent.

Dans ce cas, si vous avez fait beaucoup de modifications avant de compiler, la source de l'erreur peut être difficile à localiser et à corriger.

#### ***Therion affiche une erreur Tex bizarre***

Souvent avec plein d'instructions, disant qu'il manque des choses,...

Ce type d'erreur est toujours difficile à trouver. La encore, plus vous compilerez pendant le dessin, plus vous trouverez rapidement cette erreur.

D'après notre expérience, elle peut être due :

- soit à une erreur dans la programmation TeX qui aurait pu être faite pour définir un nouveau symbole ou la mise en page du cartouche
- soit à une erreur quelque part dans le dessin.

## **Erreur dans la programmation T<sub>E</sub>X**

En ce cas :

- Vérifiez le dernier ajout que vous avez fait, à la rigueur, commentez-le et déboguez-le.
- Vérifier que toutes les variables T<sub>E</sub>X utilisés sont déclarées. Nous avons souvent eu cette erreur à cause de la non déclaration d'une variable.

## **Erreur quelque part dans le dessin ?**

La encore, c'est notre expérience qui parle. Nous avons eu ce type d'erreur lors de l'utilisation de lignes de type pente « (slope) ». Cette ligne dessine des barbules du côté de la pente, et la longueur des barbules indique l'importance de la pente. Cette importance n'est pas calculée automatiquement, mais elle est donnée par le dessinateur dans la case « dimension-gauche (l-size) » dans l'onglet ligne du panneau de contrôle de cette ligne.

Pour ce type de ligne, si cette case n'est pas renseignée, alors, nous obtenons ce type d'erreur.

Si vous obtenez ce type d'erreur lors de l'utilisation de la ligne « pente (slope) », alors, vérifiez toutes vos lignes pente pour voir si un des points de la ligne comporte l'information l-size.

Pour aider à trouver les erreurs de dessins, *Therion* possède deux modes :

- l'option debug dans le layout
- l'option -debug pour la compilation

## **Débogage via le layout**

Dans le layout utilisé pour l'export, nous pouvons déboguer le dessin en rajoutant l'option :

`debug on`

Cette option, lors de la compilation suivante, va imprimer les étapes intermédiaires de dessin sur le document pdf final, avec un code de couleur :

- Les **lignes rouges** correspondent au dessin original, après rotation (si demandée dans le layout) et mise à l'échelle, mais non déformé. En fait c'est ce que nous voyons dans l'éditeur de dessin.
- Les **lignes bleues** : ce sont les scraps ajustés en fonction de la position des stations, avant les jointures de scraps
- Les **points jaunes** sont la position originale des points « stations topos ».
- Les points noirs reliés par des lignes noires montrent la position finale des points avec la distorsion maximale.
- Les **points orange** montrent les points avec la distance maximale dans la transformation.
- Les **lignes jaunes** entre **les points jaunes** et noirs montrent le déplacement de ces points.
- Les noms des stations sont indiqués.
- Les noms des scraps sont indiqués. **Si le nom est écrit en orange**, c'est qu'un warning (probablement une ligne de paroi qui se recoupe, ou une paroi ouverte vers l'extérieur non fermée) est associé à ce scrap.

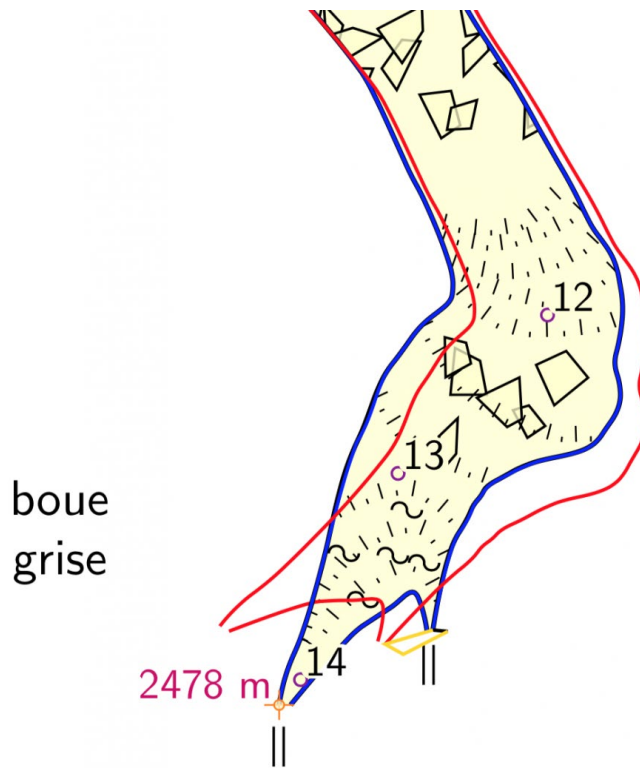


Figure 16 : l'option 'debug on'

Résultat de l'option 'debug on' sur une topographie pour laquelle nous avons artificiellement modifié la distance et l'azimut de la dernière visée 13 à 14. Noter le dessin original, la déformation et les vecteurs de déformation.

Cette commande debug peut prendre plusieurs options :

- debug on : mode debug
- debug off : pas de mode debug (par défaut, si la commande n'est pas écrite dans le layout)
- debug all : affiche toutes les informations de débogage
- debug first : affiche le dessin après la première étape de transformation
- debug second : affiche le dessin après la seconde étape de déformation
- debug scrap-names : affiche le nom des scraps
- debug station-names : affiche le nom des stations

### ***Débogage avec l'option de compilation***

Nous pouvons aussi utiliser la compilation dans un mode de débogage. Pour cela, dans le panneau de contrôle de la fenêtre de compilation, dans la case option, rajouter `-d` puis compiler.

A ce moment, dans le dossier de travail, *Therion* crée un dossier thTMPDIR. Dans ce dossier se trouvent les différents fichiers metapost, organisés par leur ordre d'utilisation. Cela peut servir pour trouver les erreurs metapost. Ce n'est pas évident à utiliser sans avoir de fortes connaissances en metapost.

Dans ce mode de compilation, le dossier thTMPDIR doit être effacé manuellement.

## **9. PENSER BASE DE DONNÉES TOPOGRAPHIQUE ET SIG**

### **▪ Définition**

Une map (carte *Therion*), c'est un groupement de scraps (morceaux de calque *Therion*), ou d'autres maps dans la même projection (plan, coupe développée ou coupe projetée).

## ▪ Comment les construit-on ?

Pour les construire, dans un fichier de données .th (dans un bloc survey ou non !), il suffit de construire un bloc

```
map <identifiant> [<options>]
# es définitions de map
Endmap
```

L'<identifiant> est le nom (sans espaces ni accents) de la map, que nous utiliserons pour l'appeler plus tard, dans un fichier *thconfig*, ou dans une autre map. Il peut donc être utile de donner un nom qui permettra de la trouver facilement. Nous pouvons, par exemple, nous contraindre à suivre des règles comme par exemple :

- le nom commence par M pour dire que c'est un nom de map
- Ensuite, nous écrivons P ou C pour indiquer que c'est une map pour Plan ou pour une Coupe,
- Nous pouvons donner le nom de la grotte
- suit du nom du réseau

Ce qui donne pour le plan du réseau Anou2 du gouffre Jean-Bernard « [MP-JB-Anou2](#) »

A l'intérieur de ce bloc, nous allons donner les scraps, ou les autres maps (oui, nous pouvons les emboîter) qui vont appartenir à cette map [MP-JB-Anou2](#).

Une map peut prendre trois options :

- l'option « -title << titre de la map >> » pour donner un titre à la map. Si ce titre est donné, alors, c'est lui qui va être affiché comme titre du pdf si nous utilisons le système de maps
- l'option « -proj <plan/extended/elevation/none> » pour indiquer la projection de la map. C'est nécessaire si les objets donnés dans la map sont des surveys et non des maps ou des scraps
- l'option « -survey <nom\_de\_la\_survey> » pour relier une map à une survey. Dans ce cas, les statistiques exportées à partir de cette map seront les statistiques de la survey associée.

## ▪ À quoi ça peut bien servir ?

Cela permet d'organiser notre dessin, à l'échelle d'un réseau, d'une cavité, voir d'un système karstique complet, et de pouvoir exporter facilement qu'une partie de la topographie.

Par exemple, nous pourrions définir les maps suivantes :

```
map MP-JB-Anou2 -title "Réseau Anou2"
  SP-anou2-1
  SP-Anou2-2
endmap
map MP-JB-Mais -title "G. du Mais"
  SP-Mais
endmap

map MP-JB-total -title "Gouffre du Jean-Bernard"
  MP-JB-Mais
  break
  MP-JB-Anou2
endmap
```

Si nous utilisons les map MP-JB-Anou2 ou MP-JB-Mais (qui appellent ici des scraps), alors, nous ne sélectionnons que cette partie du réseau du JB. En revanche, si nous sélectionnons la map MP-JB-total, alors, nous allons travailler, ici, sur les deux maps à la fois (par exemple, nous afficherons le plan des dessins définis dans ces deux maps, à savoir les trois scraps SP-Mais, SP-Anou2-1 t SP-Anou2-2)

### **La commande *break***

Dans la dernière map définie ici, la commande *break* sépare les deux maps appelées. Cela implique que ces deux maps ne seront pas sur un même niveau. Nous pouvons alors jouer sur les niveaux de superposition, ou sur des décalages visuels

### **La commande *preview* : gérer les ordres de superposition**

Cette commande ne fonctionne que si les objets appelés par la map sont des maps eux même. Reprenons notre exemple de la map MP-JB-total ci-dessus. Si nous voulons imposer la map MP-JB-Mais d'être dessinée en dessous de la map MP-JB-Anou2 (par exemple parce que nous avons mal géré la construction des scraps...), alors, il suffit d'utiliser la commande « *preview* » :

```
map MP-JB-total -title "Gouffre du Jean-Bernard"  
    MP-JB-Mais  
    break  
    MP-JB-Anou2  
    preview above MP-JB-Mais  
endmap
```

la commande « *preview* » peut prendre les arguments « *above* » (au dessous) ou « *below* » (en dessous, suivit de l'identifiant (nom) de la map à laquelle on se réfère. Ici, nous demandons à MP-JB-Anou2 d'être au dessus de MP-JB-Mais

### **Décaler visuellement des bouts de topographies**

Dans les exemples précédents, nous avons systématiquement laissés les dessins à leur position dans le réseau. Si pour des raisons de clarté, il nous faut décaler des bouts de topographies parce qu'ils se superposent et sont illisibles, nous allons utiliser les maps.

Reprenons l'exemple ci-dessus. Nous voulons maintenant que la map MP-JB-Anou2 soit décalée. Nous modifierons la map ainsi :

```
map MP-JB-total -title "Gouffre du Jean-Bernard"  
    MP-JB-Mais  
    break  
    MP-JB-Anou2 [0 15 m] above  
endmap
```

Ici, nous disons que nous voulons décaler la map MP-JB-Anou2 de 0 m en x, mais de 15 m en y, et que nous voulons qu'elle soit au dessus de la map MP-JB-Mais. Visuellement, nous verrons en plan le dessin de la galerie du Mais, avec un ombrage correspondant au réseau Anou2. La topographie en plan de ce réseau sera positionné 1.5 cm au dessus à la verticale de l'ombrage (si l'échelle d'export est au 1/1000). Si dans les scraps appartenant à la map MP-JB-Anou2 contiennent des points « Connexion de maps (map-connection) », alors, à leur position, nous verrons une ligne pointillée reliant le point sur la topographie décalée, et le point sur l'ombrage.

Sur ce site internet, un rendu d'utilisation de cette fonction peut être trouvé sur la coupe du réseau de la Combe aux Puaires, au niveau du méandre Billy.

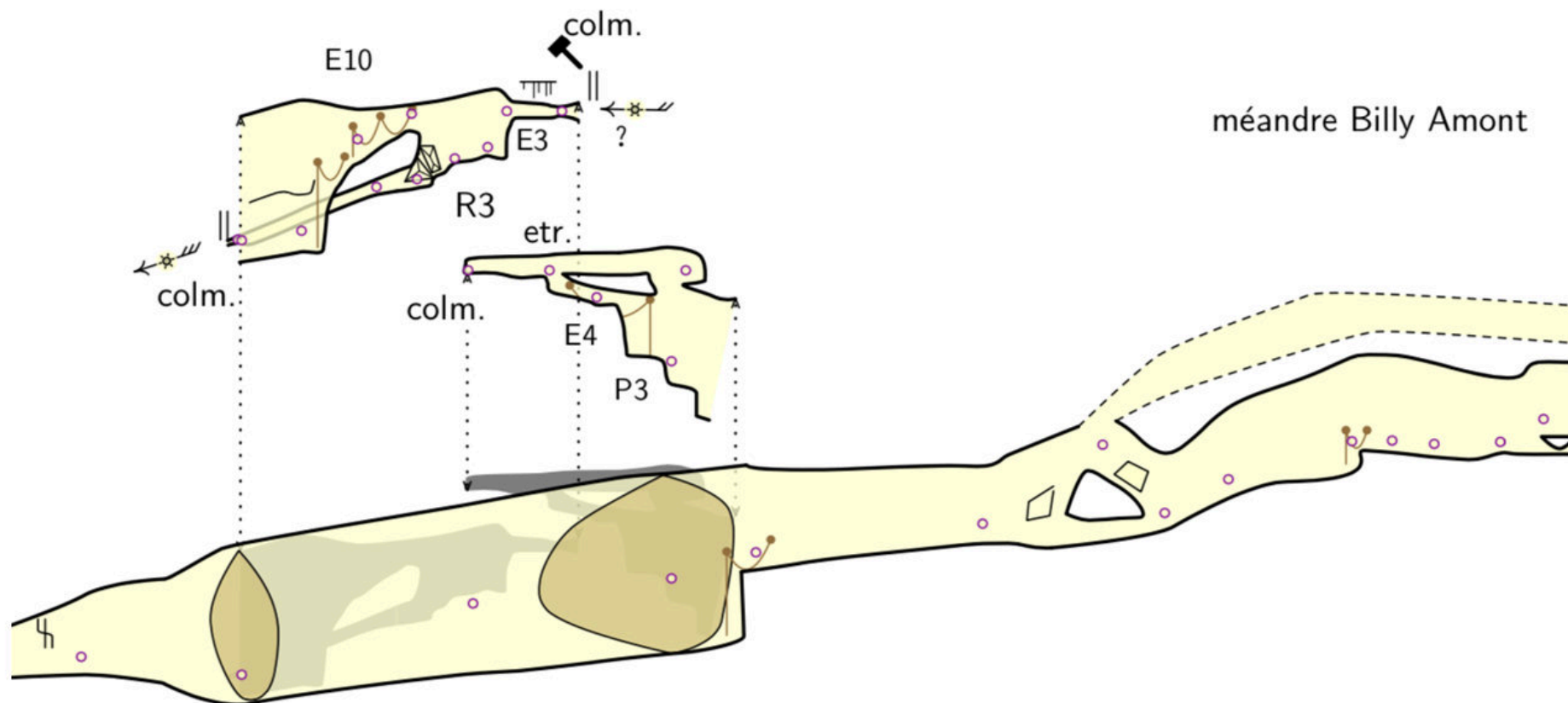


Figure 17 : Exemple de décalages (Réseau de la Combe aux Puaires, Samoëns, France).

Là encore, nous n'avons décalé que des maps, nous ne pouvons pas décaler des scraps uniquement. Si nous voulons décaler un scrap seul, alors, il suffit de créer une map qui ne contient que ce scrap.



## ▪ Comment les utiliser ?

Ah oui, c'est bien joli, nous avons défini plein de maps, mais pour qu'elles soient actives (et leurs options avec), il nous faut le sélectionner.

Cette sélection se fait dans le fichier de compilation *.thconfig*, en rajoutant avant les commandes d'exports, la commande :

```
select MaMap
```

Si la map MaMap est dans un fichier de données en dehors de tout bloc survey – endsurvey, alors, la commande si dessus est bonne.

En revanche, si la map MaMap est contenue dans une topographie (par exemple MaSurvey), alors, il faudra l'indiquer, sinon *Therion* dira qu'il ne trouve pas MaMap, et il fera comme si elle n'existait pas. La commande devra être :

```
select MaMap@MaSurvey
```

Plus complexe, si MaMap est définie dans une topographie MaSurvey, mais que cette topo est dans une autre topo (par exemple SurveyPrincipale), et que le *thconfig* appelle toutes les topographies de niveaux supérieur, alors, il nous faudra appeler Ma Map ainsi :

```
select MaMap@MaSurvey.SurveyPrincipale
```

Une partie des notions abordées ici est développée dans [l'article sur la base de données topographiques du massif du Folly pour l'Echo des Vulcains n°77](#).

Vous pouvez vous référer à [la base de données topographiques en ligne du massif du Folly](#) pour avoir accès à un exemple de structure et de fichiers de liaison.

Pour travailler efficacement avec *Therion*, surtout si nous travaillons sur une cavité complexe ou si nous pensons déjà travailler à l'échelle d'un réseau karstique ou d'un massif, il est important de bien structurer sa base de données topographiques, que ce soit à l'échelle de la petite cavité, du bout de cavité, d'une cavité complexe, ou d'un système karstologique complet.

Et non seulement il faut bien penser la structure de la base de données topographiques, mais aussi, il peut être intéressant d'appliquer certaines conventions à définir, notamment sur les noms des objets *Therion* comme les scraps, surveys et maps.

### ***Structure de la base de données topographiques d'une cavité simple***

Pour une cavité simple, nous pourrions nous contenter d'avoir un dossier spécifique pour cette cavité, avec tous les fichiers *Therion* d'entrée (les données, le dessin, la configuration et le *thconfig*) et les exports de *Therion*. Seulement, cela devient vite le bazar... et d'y retrouver n'est parfois pas toujours simple.

Il peut donc être bon de créer un dossier uniquement pour les données, par exemple nommé « Data/ », et un dossier uniquement pour les exports, que nous pourrions nommer « Ouputs/ ».

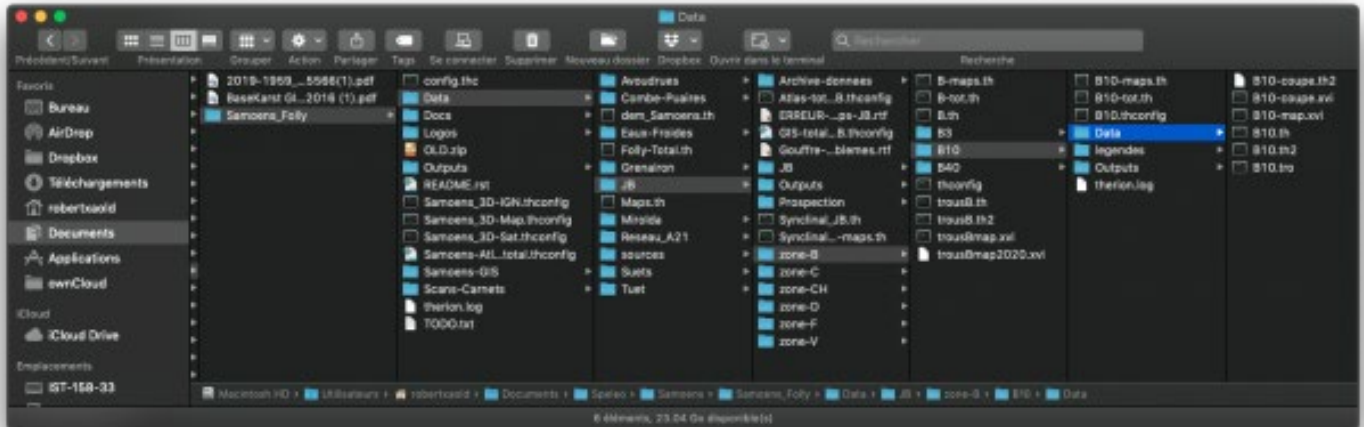
Le fichier de compilation restera dans le dossier de la grotte, et il appellera les fichiers d'entrées :

```
input Data/mesdonnees.th
input Data/mesdessins.th2
```

Nous redirigerons aussi les exports. Pour l'export xvi, il peut être intéressant de les exporter vers le dossier Data/ pour qu'ils soient avec les dessins :

```
export map -proj -plan -layout xviexport -o Data/xviplan.xvi
export map -proj -plan -layout monlayout -o Outputs/dessin-plan.pdf
```

Pour les vieux fichiers que nous voulons garder (par exemple les anciens exports), nous pouvons les stocker dans un dossier « OLD/ ».

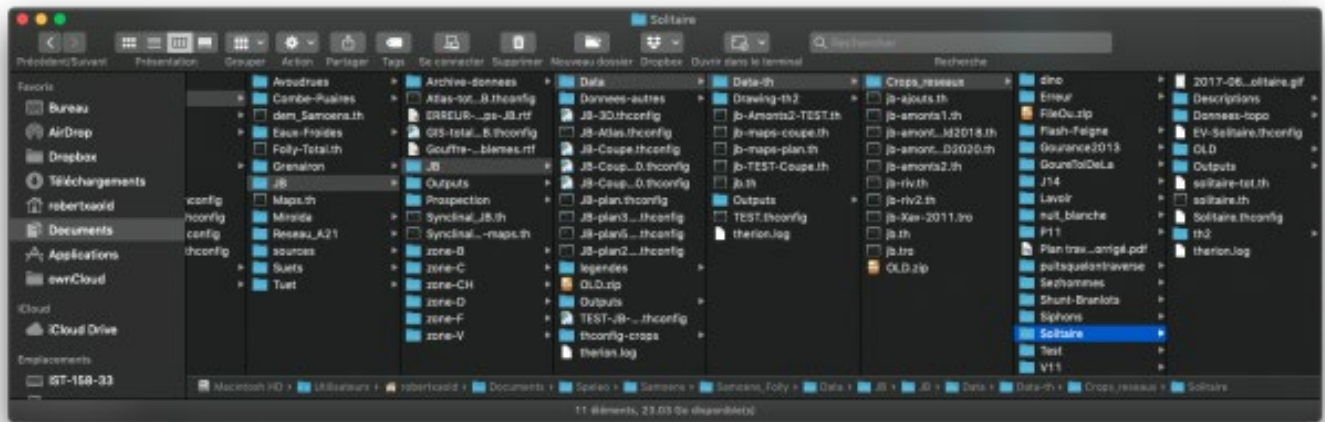


Structure d'une base de données topographique, avec ici le détail pour la cavité Simple B10 qui appartient à la zone B du système du gouffre Jean Bernard, dans le massif du Folly.

### **Structure de la base de données topographiques d'une cavité complexe**

Pour une cavité complexe, le travail sera facilité si nous découpons la cavité en plusieurs bouts. Chaque bout aura donc son propre dossier, de même structure que pour une cavité simple, avec les dossiers « Data/ » et « Outputs/ ». En amont de chaque dossier de parties de la cavité, nous rajouterons un fichier .th qui va appeler toutes les données partielles (topographies et dessins), et les relier entre elles. Dans l'exemple suivant, c'est le fichier « jb.th » dans « JB/Data/ ».

Nous pouvons aussi rajouter, au même endroit, un ou des autres fichiers pour gérer la déclaration des maps. Dans l'exemple suivant, ce sont les fichiers « jp-maps-plan.th » et « jb-maps-coupe.th ».



Structure de la base de données pour une cavité complexe. Ici, par exemple, le gouffre Jean-Bernard (Samoëns, 74, France).

### **Structure de la base de données topographiques d'un système karstique**

Pour un système karstique, il suffit de faire la même chose que pour une cavité complexe, sachant que les différents dossiers seront alors les dossiers des cavités composant le système karstique.

La encore, nous devons rajouter un fichier .th d'appel et de jointure des topographies des cavités. Normalement, le travail fait au niveau inférieur (au niveau de chaque cavité) n'est pas à répéter, il suffit d'appeler les fichiers maîtres de chaque cavité (qui comportent tout ce qu'il faut, que ce soit les données topographiques, les dessins et les jointures).

Il faudra aussi rajouter un système de gestion des maps, soit dans le fichier .th d'appel des données, soit dans un fichier dédié.

C'est là où nous voyons l'intérêt de la hiérarchisation pour pouvoir appeler les objets. Par exemple, nous voulons appeler une map « MP-x-zazou » qui est déclarée dans la partie (survey) « fond » de la cavité (survey) « x ». Cette cavité est dans la zone (survey) « zoneC » du système karstique. Alors, en étant au niveau du système karstique, si nous avons besoin d'appeler cette map, alors nous écrivons :

```
select MP-x-zazou@fond.x.soneC
```

Dans l'appel, nous lisons immédiatement le sens d'imbrication.

- **Conventions pour nommer les objets et les fichiers**

Pour des raisons de clarté et de facilité de travail, respecter une convention pour nommer les différents objets *Therion* est très utile. C'est à chacun de se construire sa propre convention, mais il faut alors s'y référer.

### **Conventions pour nommer les objets Therion**

Il peut-être important qu'à la lecture du nom d'un objet, cette convention permette de savoir :

- si l'objet est un scrap, une survey ou une map

- si l'objet est associé à une projection Plan, Coupe Développée, Coupe Projetée ou None,
- si cet objet appartient à telle ou telle cavité
- si cet objet appartient à telle ou telle partie spécifique d'une cavité.

Pour cela, nous pouvons, par exemple, construire le nom d'un objet en suivant cette convention :

`<type><proj>-<cavité>-<réseau>`

où :

- `<type>` sera le type d'objet :
  - M pour map,
  - S pour scrap
  - T pour survey
- `<proj>` sera la projection
  - P pour plan
  - C pour coupe développée
  - CP pour coupe projetée
  - N pour none
- `<cavité>` est le nom de la cavité
- `<réseau>` est le nom de la partie de la cavité complexe

Par exemple, pour un scrap en plan du réseau « fond » du gouffre X, nous le nommerons :

`SP-x-fond`

### **Conventions pour nommer les fichiers Therion**

De la même manière, nous pouvons aussi adopter une convention pour nommer les fichiers *Therion*. Par exemple, pour la cavité XY, nous pourrions adopter le nom de fichiers :

- XY.th pour les données topographiques (dans Data/)
- XY-plan.th2 pour le dessin en plan (dans Data/)
- XY-coupe.th2 pour le dessin en coupe (dans Data/)
- XY-plan.xvi et XY-coupe.xvi pour les fichiers xvi (dans Data/)
- XY-tot.th pour le fichier de liaison des données topographiques et de dessin (dans le dossier maître de la cavité)
- XY-maps.th pour le fichier de définition des maps dans le dossier maître de la cavité
- XY.thconfig pour le fichier de compilation dans le dossier maître de la cavité
- XY-plan.pdf, XY-coupe.pdf, XY.loch... pour les fichiers exportés dans le dossier Outputs/

Dans le cas de cavités qui sont susceptibles d'évoluer dans le temps, il peut être intéressant de rajouter à la fin du nom la date (a minima l'année) pour les fichiers xvi et les autres exports.

L'avantage de suivre ce type de convention, c'est que pour les personnes qui travaillent avec un Terminal, il est facile d'appliquer des modifications globales extrêmement rapidement, au lieu de toucher chaque fichier un à un !

Evidemment, ceci n'est qu'un exemple, qu'il faut adapter chacun à ses besoins.

Lorsque nous travaillons sur une cavité complexe, voir un système karstologique, nous travaillons généralement sur plein de petits bouts séparés. Il se peut aussi que pour que le rendu des dessins corresponde à ce que nous désirons, nous ajoutons beaucoup de commandes de configuration dans nos

layouts à l'intérieur de nos fichiers de compilation. En conséquence, ces fichiers peuvent vite devenir longs. C'est d'autant plus le cas lors que nous modifions des symboles prédéfinis et/ou que nous en écrivons des nouveaux ! Cela devient vite difficile pour trouver une ligne spécifique rapidement.

Si nous travaillons sur pleins de bouts différents, il se peut que nous ayons en plus plusieurs fichiers de compilation, avec de nombreuses commandes identiques.

Nous pouvons simplifier le tout, en créant un fichier de configuration maître que nous appellerons dans les layouts de fichiers de compilation.

Par exemple, nous pourrions copier toutes nos définitions de nouveaux points dans ce fichier de configuration. Ces définitions devront être dans un bloc layout – endlayout, que nous pourrions appeler « nouveaux\_pts » par exemple. Ce fichier de configuration est créé dans le dossier le plus en amont.

Dans les fichiers de compilation, nous allons appeler la définition de ces nouveaux points en deux étapes.

La première, nous devons importer le fichier de configuration (ici appelé config.thc, et situé deux dossiers au dessus du dossier contenant le fichier de configuration que nous travaillons), avec la ligne :

```
input ../../config.thc
```

Ceci permet de dire à *Therion* qu'il va falloir aller chercher des informations dans le fichier config.thc. Mais ce n'est pas suffisant.

il faut aussi dire dans le layout de mise en page du fichier de compilation, que nous voulons utiliser telle ou telle définition du fichier de configuration config.thc. Par exemple dans le layout de configuration :

« MonLayout », si nous voulons utiliser les nouveaux points que nous avons définis dans le layout « nouveaux\_pts » du fichier de configuration config.thc, alors, il faudra utiliser la commande copy :

```
input MaTopo.th # Importer les données topo les dessins, les maps
input ../../config.thc # Importer le fichier de configuration

layout MonLayout # Début du layout MonLayout
  copy nouveaux_pts # Appelle le layout nouveaux_pts du fichier config.thc
  .... # autres commandes du layout
endlayout

select MaMap@MaSurvey
export map -proj plan -layout MonLayout -o MaGrotte.pdf
```

Vous l'avez compris, ainsi, vous pouvez simplifier à l'extrême vos fichiers de compilation ! Vous pouvez ainsi faire des layouts spéciaux, bien définis dans un fichier de configuration globale, et ainsi les appeler en fonction des besoins dans les fichiers de compilation.

La base de données topographique du massif du Folly utilise ce système là.

Comprendre ce qu'est un système de coordonnées géographiques, ce qu'est un référentiel, ce qu'est un ellipsoïde et ce qu'est une projection est la base de tout travail avec des données géoréférencées, notamment avec des logiciels SIG.

Une recherche sur internet sur les systèmes de coordonnées géographiques et les projections conduit à de nombreuses pages de cours et d'informations la dessus, comme par exemple [la page Wikipedia](#) ou celle de [l'IGN](#).

Cette page ne va pas reprendre toutes les notions, mais uniquement en reprendre les points importants.

Il faut bien comprendre que

- la terre est ronde ; Nous utilisons souvent pour nous repérer des coordonnées en dimensions, ce sont les coordonnées géographiques latitude / longitude.
- Souvent, ce n'est pas spécifié (et c'est une erreur !), mais pour pouvoir nous repérer, malgré les coordonnées, il nous faut une référence : nous sommes aux latitude/longitude, mais par rapport à quel point (0,0) ? Généralement, les GPS de randonnées ou des téléphones portables donnent les coordonnées latitude / longitudes dans le référentiel WGS84.
- et que les plans et les coupes que nous produisons sont en 2 dimensions. Il faut donc effectuer une projection, qui est une fonction mathématique de transformer des coordonnées en 3D sphériques sur une surface en 2D.
- Il existe plusieurs types de projections, c'est à dire de fonction mathématique. Pour cela, il faut connaître le type de cette projection (et donc l'indiquer sur les cartes et dans les tableaux de données). La encore, il faut savoir à quel référentiel nous nous référons, il faut aussi l'indiquer.
- A cela, il nous faut rajouter l'information de quelle ellipsoïde de référence nous utilisons.

Les logiciels de Système d'Information Géographique (SIG) sont des logiciels très complexes, et qui peuvent être déroutants pour les débutants avec ces logiciels. Ceci parce que ce sont des logiciels :

- qui traitent de données géoréférencées, c'est à dire qui possèdent un référencement géographique. Il est donc indispensable de connaître les définitions de base autour des systèmes de coordonnées et de projections.
- qui travaillent avec plusieurs types de données :
  - des données dites rasters : grosso modo, ce sont des images. En fait, une image est un raster, mais tout type de grille géoréférencée est un raster. C'est une description d'une grille de pixels
  - des données dites vectorielles : les fichiers shapefiles en font partis. Ils peuvent contenir des points, des lignes et des polygones. Chaque entité est géoréférencée, et à chaque entité, différents champs peuvent être associés.
  - des bases de données complexes
- qui travaillent sur des fichiers qui peuvent être très très lourds (plusieurs Go)
- qui permettent des opérations mathématiques sur les objets ci-dessus (par exemple, projeter un raster ou un vecteur)
- qui permettent la mise en page de cartes complexes

Les logiciels SIG probablement les plus connus sont payants (et chers...), ce sont *ArcGIS* ou *Globalmapper*. Il en existe des libres, comme [Grass](#) ou [QGIS](#), qui sont assez complets, ou d'autres qui sont dédiés à des applications spécifiques, souvent en lien avec l'hydrologie. Pour les inconditionnels du terminal et des commandes en ligne, beaucoup de traitement peuvent se faire avec la bibliothèque [GDAL](#).

## LES TYPES DE DONNÉES

Que ce soit un raster ou un vecteur, chaque élément utilisé dans un SIG sera considéré comme une couche de données. Dans la pratique, visuellement, chaque couche correspond à et fonctionne comme un calque.

## **Données rasters**

Les rasters sont des fichiers géoréférencés décrivant une grille. A chaque point de la grille, une valeur est associée.

Une image, est un raster : à chaque pixel (point de grille), est associée une valeur qui correspond à la couleur du pixel.

Un modèle numérique de terrain (MNT) est aussi un raster : a chaque point de la grille est associée une valeur qui correspond à l'altitude de ce point.

En fait, nous pouvons créer autant de type de rasters que nous voulons, il suffit de savoir à quoi correspond la valeur associée aux nœuds de la grille.

Il y a beaucoup de formats différents pour un raster. Cela peut être un simple jpg (mais à ce moment, il n'y a pas de géoréférencement), un fichier texte, un fichier GeoTIFF,... Comme ces fichiers peuvent être très gros, ils sont la plupart du temps sous forme compressée.

## **Données Vecteurs**

Les fichiers vecteurs les plus classiques sont les fichiers shapefile. Ils vont contenir des objets vecteurs géoréférencés, qui peuvent soit être des points, de la ligne ou des polygones.

Pour chaque entité (point, ligne, vecteur), un ou des attributs sont associés. Pour chaque shapefile, c'est en fait un tableau, où chaque ligne correspond à une entité du shapefile, et chaque colonne aux différents attributs. A minima, il y a leur identifiant et leurs coordonnées. Les attributs peuvent être des nombres (valeurs, réelles ou entières), de booléens (de valeur vrai ou faux) ou des chaînes de caractères.

Nous pouvons donc faire des liaisons entre une base de données géoréférencée et des shapefile.

### **▪ Fonctionnement d'un logiciel SIG (à interface graphique)**

Pour ce paragraphe, nous prendrons des exemples avec le logiciel libre et gratuit QGIS.

#### ***Un mode de travail et un mode de mise en page***

Classiquement les logiciels SIG possèdent deux modes de travail :

- Un premier pour travailler sur notre projet, c'est à dire importer les couches que nous voulons, choisir les éléments que nous voulons voir, leur couleur, les trier, faire des calculs complexes,...
- Un second pour effectuer des mises en pages, c'est à dire pour préparer la carte finale que nous voudrions exporter ou imprimer. C'est notamment ici, que nous formatons et choisissons la taille du papier, l'emplacement de la figure/carte sur le papier, le rajout d'échelle, de graticules, de coordonnées, de légende,...

Pour passer de l'un à l'autre, cela se fait généralement dans les onglets du projet (barre principale supérieure).

#### ***Principaux panneaux***

Dans le mode de travail, un logiciel SIG va posséder plusieurs panneaux, que nous pouvons faire apparaître ou disparaître via les menus « fenêtres ».

Le panneau principal est l'affichage visuel des couches (calques) dans la projection du projet. C'est ici que nous allons voir ce qu'il se passe lors que nous rajoutons un objet, ou que nous changeons une couleur, une échelle, un symbole...

A cela est souvent associé :

- un panneau de navigation dans les couches : il permet de sélectionner les couches (avec les légendes), les ordonner (ce sont des calques, une couche opaque au-dessus d'une autre la masquera), accéder à leur propriétés pour modifier leur apparence,...
- un panneau d'outils de traitement (toolbox) qui permet de faire plein de traitements des vecteurs et des rasters. C'est par exemple ici que nous trouverons les fonctions de reprojections.
- un panneau de résultat de recherche lorsque nous cherchons des informations sur les couches.

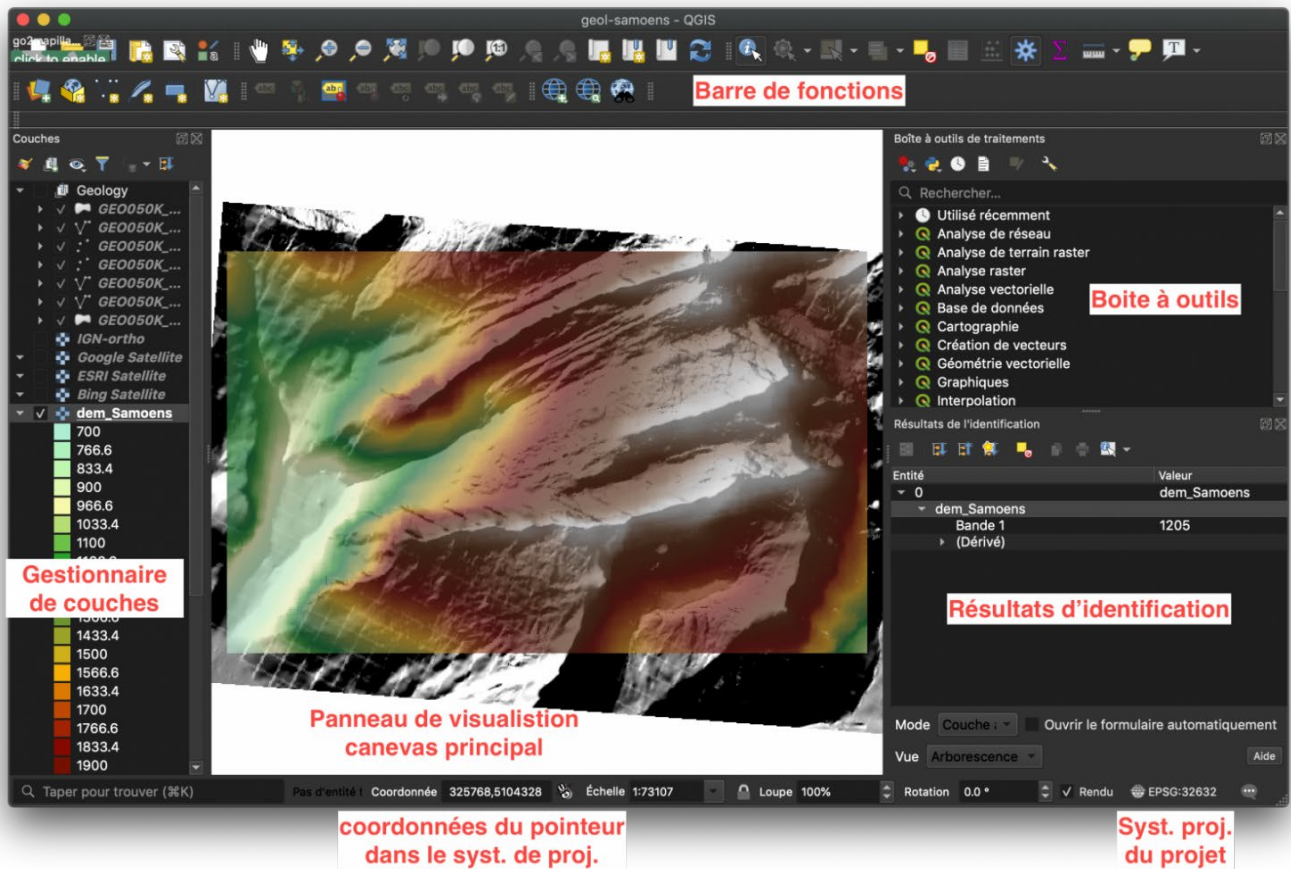


Figure 18 : Quelques panneaux classiques sous QGIS.

## Importer des couches

Pour importer des couches :

- soit il faut passer par les menus du bandeau supérieur
  - Couches → Ajouter une couche → Ajouter une couche Raster
  - Couches → Ajouter une couche → Ajouter une couche Vecteur

Ou tout simplement faire un glisser-déposé du fichier à importer dans le panneau de gestion des couches.

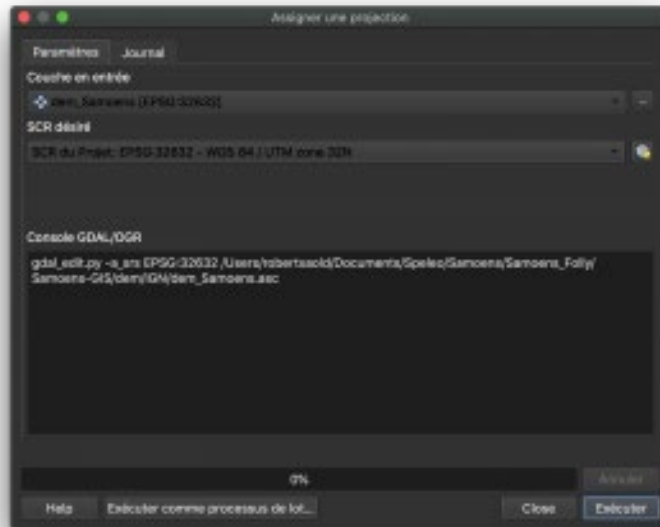
Pour modifier l'ordre des couches, il suffit de changer l'ordre à la souris.



Pour modifier l'apparence de la couche, il faut aller dans les propriétés de la couche, soit en double-cliquant sur le nom de la couche, soit en faisant un clic-droit → propriétés sur le nom de la couche.

Si la couche importée ne comporte pas de renseignements sur sa projection, alors, il faut lui assigner une projection pour pouvoir la voir au bon endroit dans le logiciel SIG.

Pour cela, pour les rasters, utiliser l'outil Raster → Projection → Assigner une projection, remplir les champs et exécuter l'outil.



L'outil assigner une projection Raster. Nous voyons en bas que l'outil utilisé sera l'outil issu de la bibliothèque GDAL.

### ***Faire des opérations***

Pour faire des opérations, il nous faut encore utiliser la boîte à outils et chercher l'outil qui va nous permettre de faire l'opération que nous désirons.

Par exemple, si nous désirons projeter un Raster, nous pouvons directement aller dans le menu Raster → Projection → Projeter. Nous pouvons aussi, dans la boîte à outils, taper « projection » dans le champ de recherche, et choisir l'outil « Projection (warp) ». Il nous reste à remplir les champs, et exécuter.

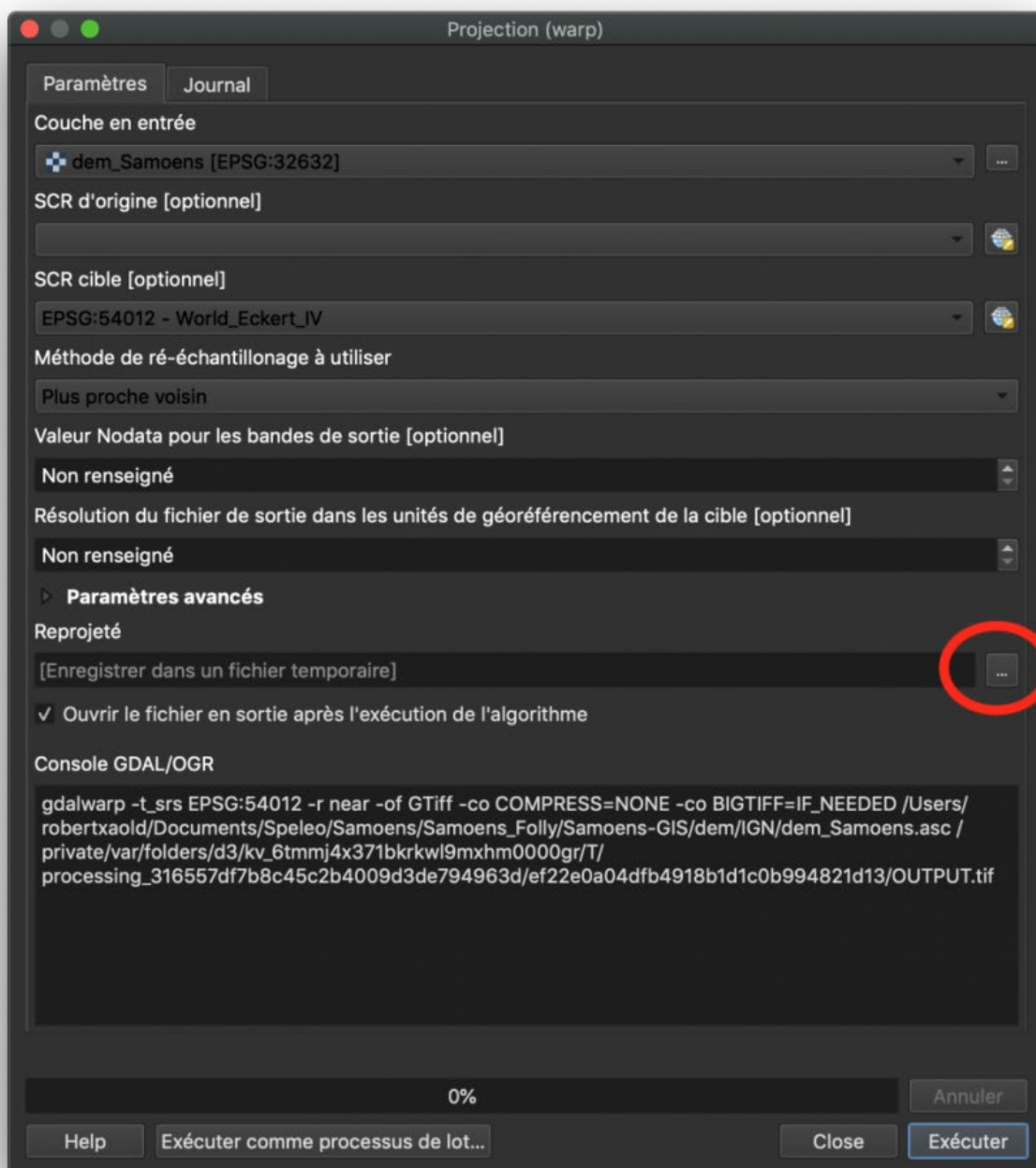


Figure 19 : Outil de reprojection d'un raster (wrap).

Le rond rouge montre où il faut cliquer pour choisir où nous voulons enregistrer le fichier projeté

### ***Faire une mise en page***

Pour faire une mise en page, il faut passer en mode mise en page. Sous QGIS, nous devons le faire en allant dans Projet → nouvelle mise en page

Une fois la mise en page créée, une page blanche s'ouvre. A l'endroit où nous voulons la carte, il faut créer le cadre de la carte avec le bouton « nouvelle carte ». Ensuite, il faut jouer avec les paramètres de mise en page, dans le panneau de droite et dans les menus pour finaliser la mise en page.

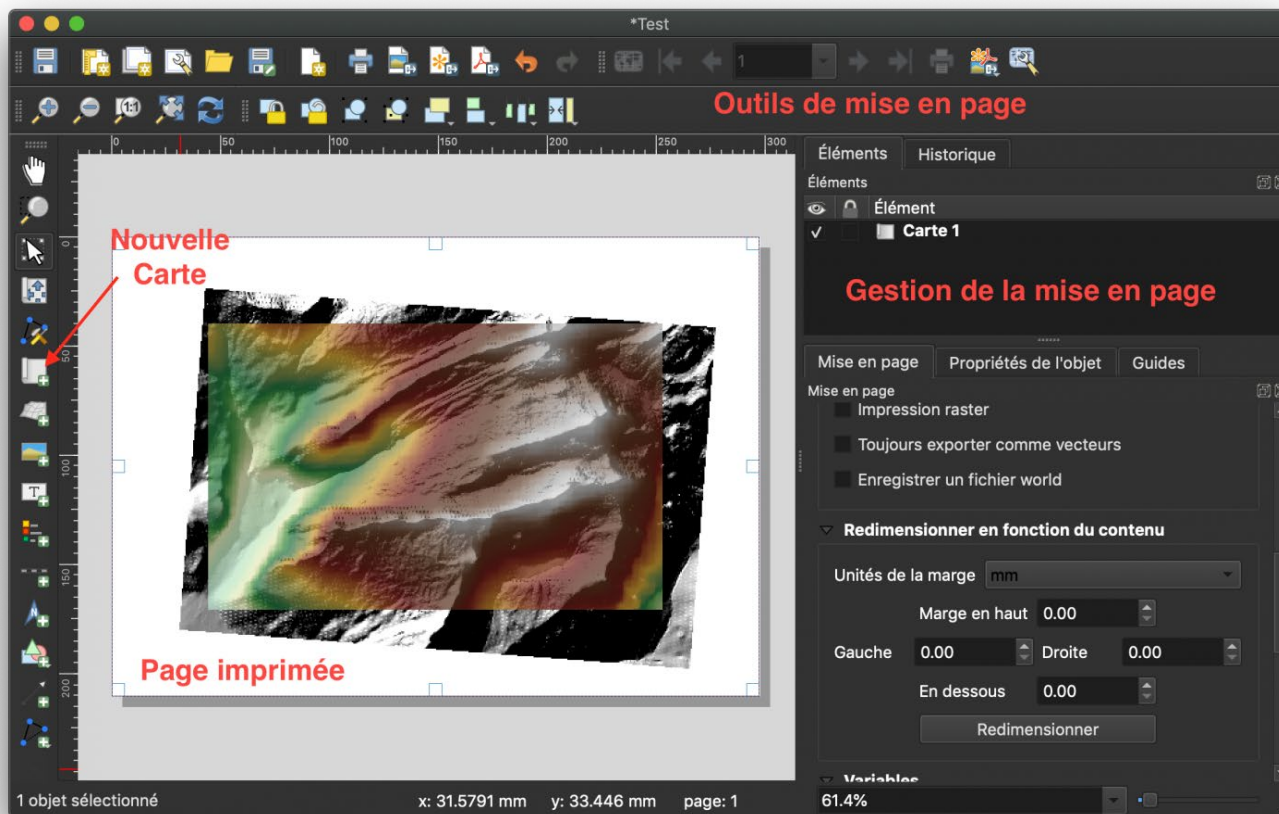


Figure 20 : Mode Mise en Page de QGIS avec une carte en cours de travail.

### Quelques règles de base

Pour travailler avec un logiciel SIG, il faut faire attention à quelques points, histoire de ne pas avoir de mauvaises surprises :

- Soyez très rigoureux dans l'organisation de votre projet SIG sinon, votre dossier de travail sera un bazar sans nom...
- Attention si vous travaillez sur des données lourdes, vous allez consommer très rapidement de l'espace disque sur votre ordinateur, soyez-en conscient (en l'espace d'une après midi de travail, il est possible de consommer 10, 20, 50 Go de votre disque dur...). Aussi, un projet avec des fichiers lourds va demander l'utilisation de beaucoup de mémoire vive. Si votre ordinateur n'a pas une quantité suffisante de mémoire vive, alors, soit les opérations deviendront extrêmement lentes, soit les opérations renverront une erreur mémoire.
- Aussi, sachez dans quelle projection est votre projet. Maîtrisez cette projection, projetez toutes les couches dans cette projection pour vous éviter des surprises par la suite...

Et oui, avec *Therion*, nous pouvons ajouter le relief (Modèle Numérique de Terrain ou de Surface – MNT/MNS) au dessus de notre cavité. Mais pour cela, il faut préparer un MNT au format *Therion*.

Ainsi, sur les exports 3D, nous aurons le relief au-dessus de notre cavité. L'ajout de ce MNT permet aussi de draper une image (aérienne, satellite, carte,...) sur le MNT, et ainsi, de visualiser le modèle en 3D avec une image sur le relief. Mais plus encore, avec cette image, nous allons pouvoir la mettre en arrière-plan de la carte en plan, avec plus ou moins de transparence !

## Une approche semi-automatique : TerrainTool

C'est la solution de facilité. TerrainTool est un petit utilitaire pour télécharger un MNT correspondant à la zone que nous voulons, et le formater pour *Therion*. Cet utilitaire se trouve ici : <http://ubss.org.uk/terrainool/terrainool.php>.

Il va générer un MNT à partir de données Aster ou SRTM (à choisir), qui toutes deux sont à 30 m de résolution.

- **Construire le MNT pour *Therion* à partir de sa propre base de données de relief**

Générer soi-même un MNT pour *Therion* peut être difficile car cela demande quelques connaissances en SIG (Système d'Information Géographique). Un logiciel libre utile pour faire toutes ces étapes est [QGIS](#).

### **Les topographies en Shapefiles**

Pour nous repérer dans le logiciel SIG, nous allons avoir besoin de la position des cavités topographiées. Pour cela, il faut exporter les cavités au format shapefile, avec, dans le *thconfig*, cette commande export:

```
export map -fmt esri -layout monlayout -o MaGrotte
```

Cette commande va créer un dossier MaGrotte, avec plusieurs fichiers à l'intérieur. Les fichiers qui vont nous intéresser sont les fichiers avec l'extension .shp.

Idéalement, pour faire cet export, dans le layout monlayout, il faudrait avoir défini la projection de l'export avec la commande cs.

### **Où trouver des MNTs ?**

Aujourd'hui, il n'est pas évident de trouver librement des MNTs de qualité, pour toutes les régions du monde, avec une très bonne résolution. Les MNT ASTER et [SRTM](#) sont corrects et ont une résolution globale de 30 m, c'est à dire qu'un pixel du MNT correspond en vrai à un carré de 30 m sur 30 m.

Obtenir des MNT de meilleure résolution est difficile. En France, à condition d'y mettre le prix, il est possible d'obtenir auprès de l'IGN un MNT de 1 à 5 m de résolution.

Il faut quand même être conscient que plus la résolution est bonne, pour une surface donnée, plus les fichiers sont gros et gourmands en mémoire, temps et énergie pour les travailler.

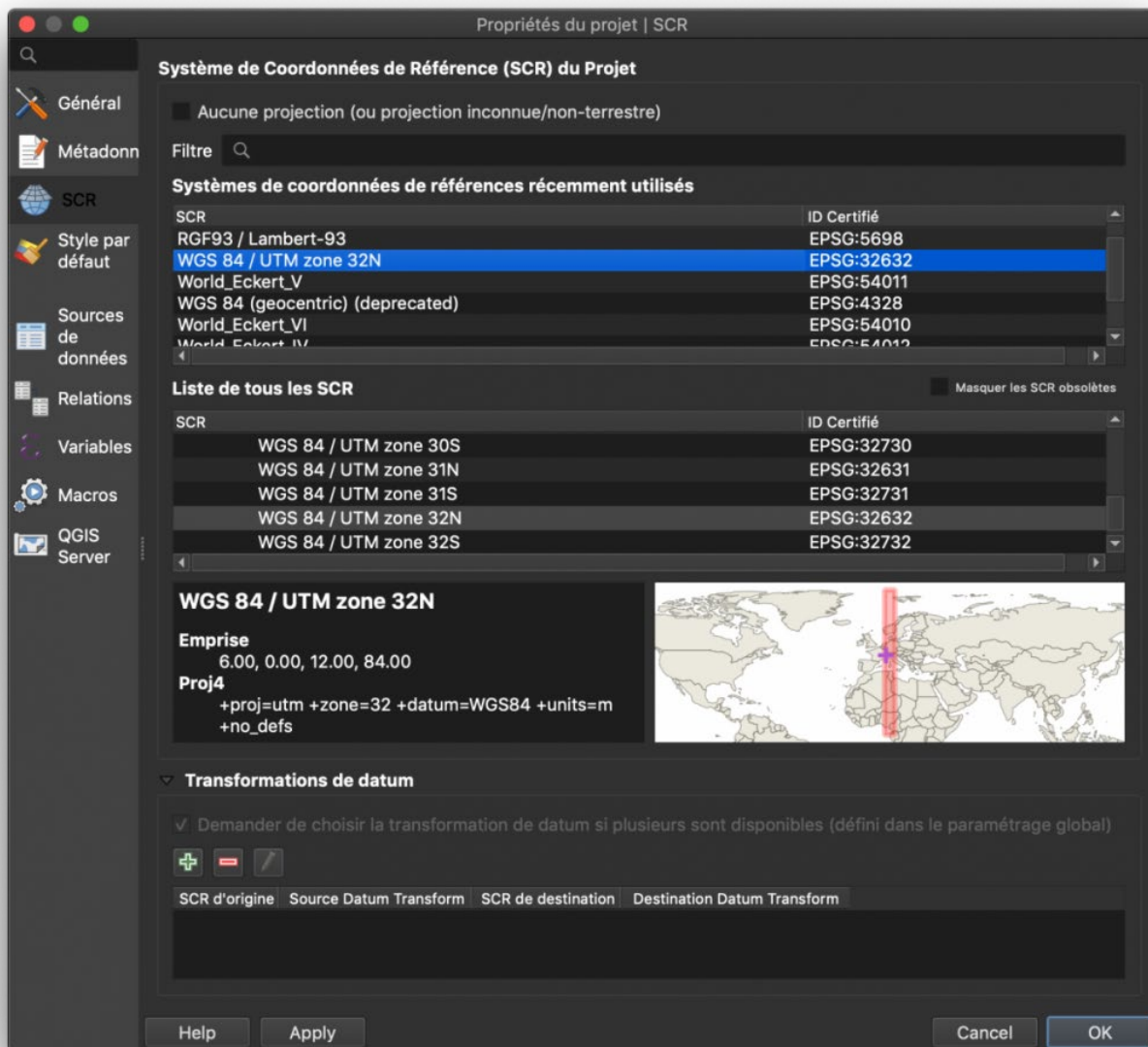
Pour le téléchargement, les fichiers GeoTIFFs sont compressés et donc moins lourds à utiliser que les fichiers Esri ASCII.

### **Charger le MNT dans un logiciel SIG (ici, QGIS)**

D'abord, ouvrir QGIS puis Créer un nouveau document. Ce sera notre projet de travail.

Il nous faut définir la projection dans laquelle nous souhaitons travailler. Pour cela, cliquer en bas à droite sur la case « EPSG:xxxxx » où xxxxx est une série de nombre correspondant au code EPSG de la projection actuelle. La liste des systèmes de projection s'affiche (liste des SCR), recherchez la projection

que vous utilisez dans votre projet topographique [la commande cs du layout] (le plus souvent, WGS84/UTM zone 30/31/32N ou un des Lambert), la sélectionner, cliquer sur Apply puis OK. En bas à droite, le numéro EPSG devrait avoir changé. Enregistrer le projet (Projet → Enregistrer Sous).



### Choix du SCR du projet sous QGIS.

Faire ensuite un glisser-déposer du MNT Téléchargé dans l'explorateur de couches, ça devrait le charger. Sinon faites Couche → Ajouter une couche Raster. Vous devriez voir votre MNT en noir et blanc.

Normalement il devrait avoir été reprojeté à la volée, vérifier quand même que la projection du projet n'ait pas changée. Mais pour que le travail soit fait proprement et pour éviter les erreurs de projection, il faut reprojeter le MNT dans le système de projection du projet. Cela se fait avec Raster → Projection → Projection (warp). Remplissez les champs non optionnels.

Une fois le MNT chargé, vous pouvez importer le fichier vectoriel shapefile [MaGrotte/outlines2D.shp](#) (vous pouvez aussi essayer les autres, mais celui-ci, c'est le contour des galeries). Si vous ne le voyez pas

apparaître ou si le logiciel QGIS se plaint, c'est que la projection du shapefile n'est pas définie. En ce cas, double-cliquer sur la couche vecteur dans le panneau de Couches. Les propriétés de la couche s'affichent, aller dans l'onglet « Source », et ici, vous pouvez configurer le système de coordonnées (il faut mettre le même que celui utilisé pour l'export, celui imposé par la commande cs du layout).

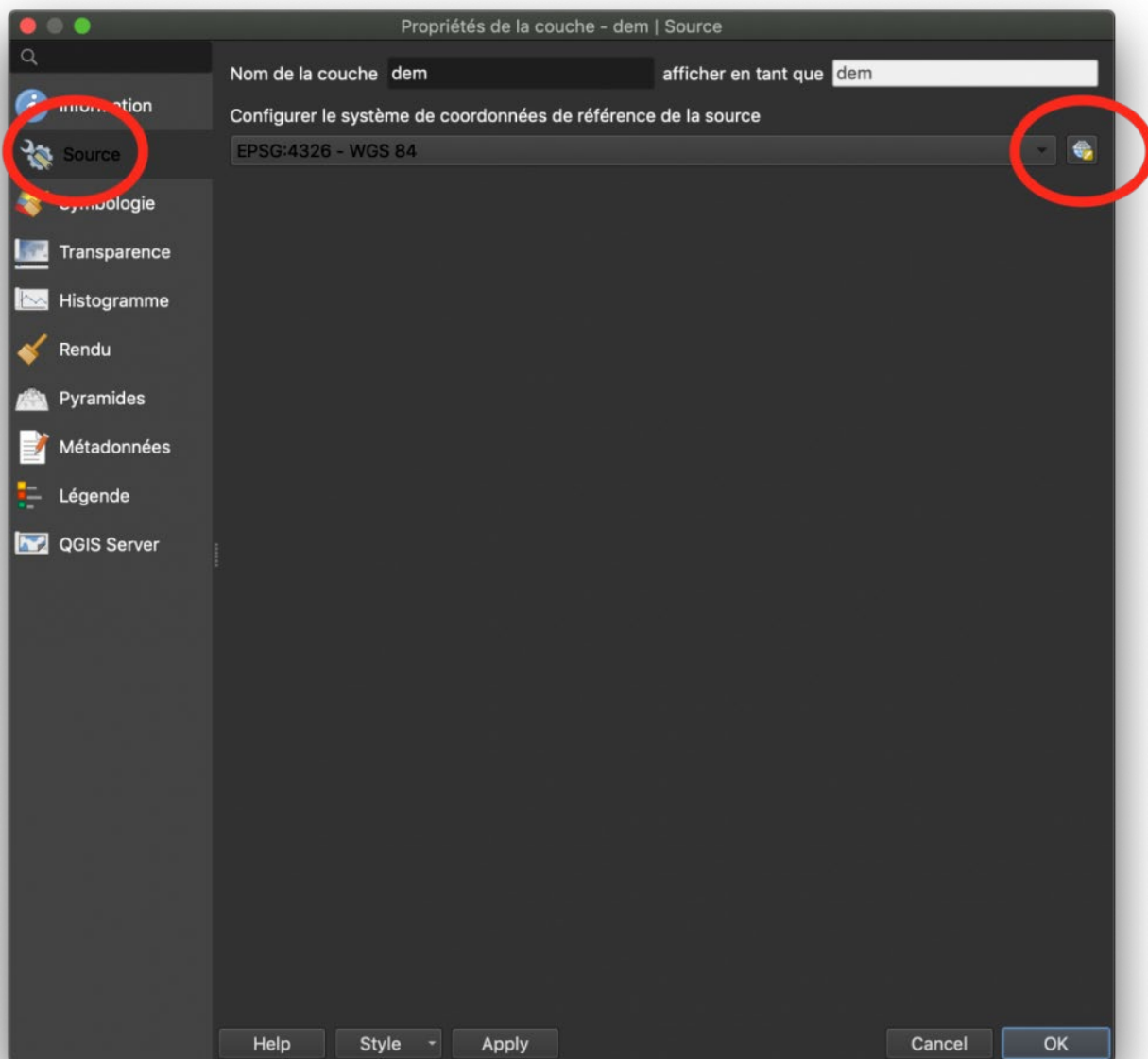


Figure 21 : Choix du système de projection du MNT.

Si le système de projection du MNT n'est pas dans le menu déroulant, alors, cliquer sur le petit bouton à gauche (cercle rouge) pour afficher toute la base de données des systèmes de projection.

Si la couche vecteur est en dessous du MNT, elle ne va pas se voir non plus. Il faut donc la remonter au dessus du MNT ( la souris, dans le panneau Couches). Pour zoomer dessus, sélectionner dans le panneau Couches le shapefile, puis faire un clic-droit → zoomer sur la couche.

## Extraire la partie du MNT qui nous intéresse

Une fois que le MNT et les trous sont visibles sur notre plan de travail, nous allons découper le MNT.

Pour cela :

- aller dans Raster → Extraction → Découper un raster selon une empreise.
- Sélectionner la bonne couche source
- cliquer sur le petit bouton à droite de la case « Etendue de découpage »
- Choisissez « Sélectionner l'emprise du canevas »
- sur la carte, cliquer sur un des coins de la zone que nous voulons, puis glisser la souris en maintenant appuyé, vers le coin opposé, la fenêtre de découpe s'affiche de nouveau avec les coordonnées de l'emprise à découper,
- A droite de la case « Découpé », cliquer sur le bouton avec les petits points → enregistrer sous et choisissez le dossier et le nom du nouveau raster
- Cliquer sur « Exécuter »

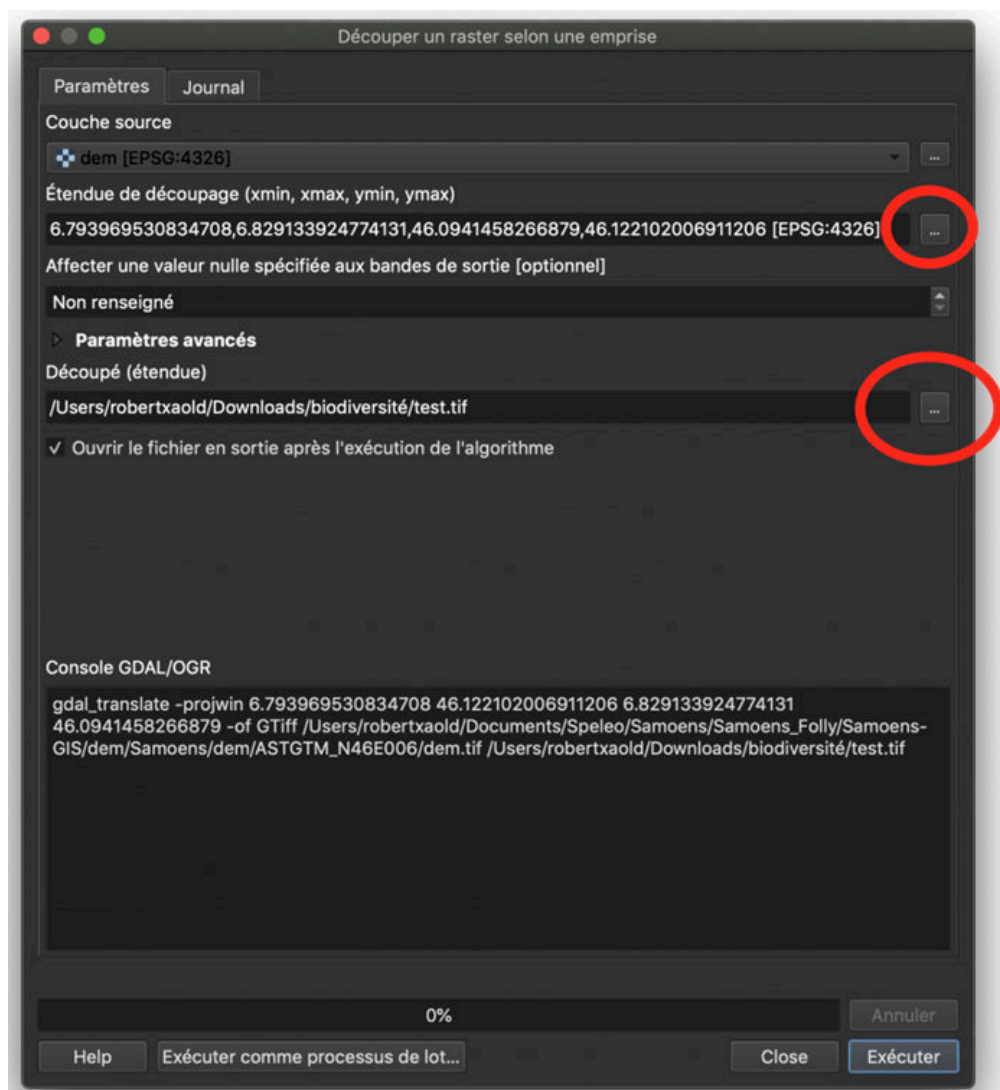


Figure 22 : Outil de découpe d'un raster sous QGIS, avec choix de la zone de découpe.

Et voilà, c'est fait ! Le nouveau raster se rajoute au projet dans les Couches. Vous pouvez enregistrer !

## Exporter le MNT

L'étape précédente a enregistré un Raster au format .tif (GeoTIFFs). C'est très bien, sauf que ce format n'est pas transformable en tant que tel pour *Therion*. Il faut donc exporter ce MNT dans le format ESRI ASCII non compressé :

- Aller dans Raster → Conversion → Convertir
- La fenêtre de conversion d'ouvre
- Choisir la bonne couche d'entrée (le MNT découpé)
- Cliquer à droite de Converti sur le petit bouton → Enregistrer dans un fichier
- Une fenêtre s'affiche avec un explorateur de fichier
- En bas, choisir le type de fichier « ASC files (\*.asc)
- Donner un nom a votre fichier (et choisir le dossier d'exportation)
- Cliquer sur Enregistrer pour revenir à la fenêtre précédente
- Cliquer sur « Paramètres avancés »
- Dans « Profil », choisir « No Compression » (= pas de compression, option fondamental pour notre utilisation)
- Exécuter

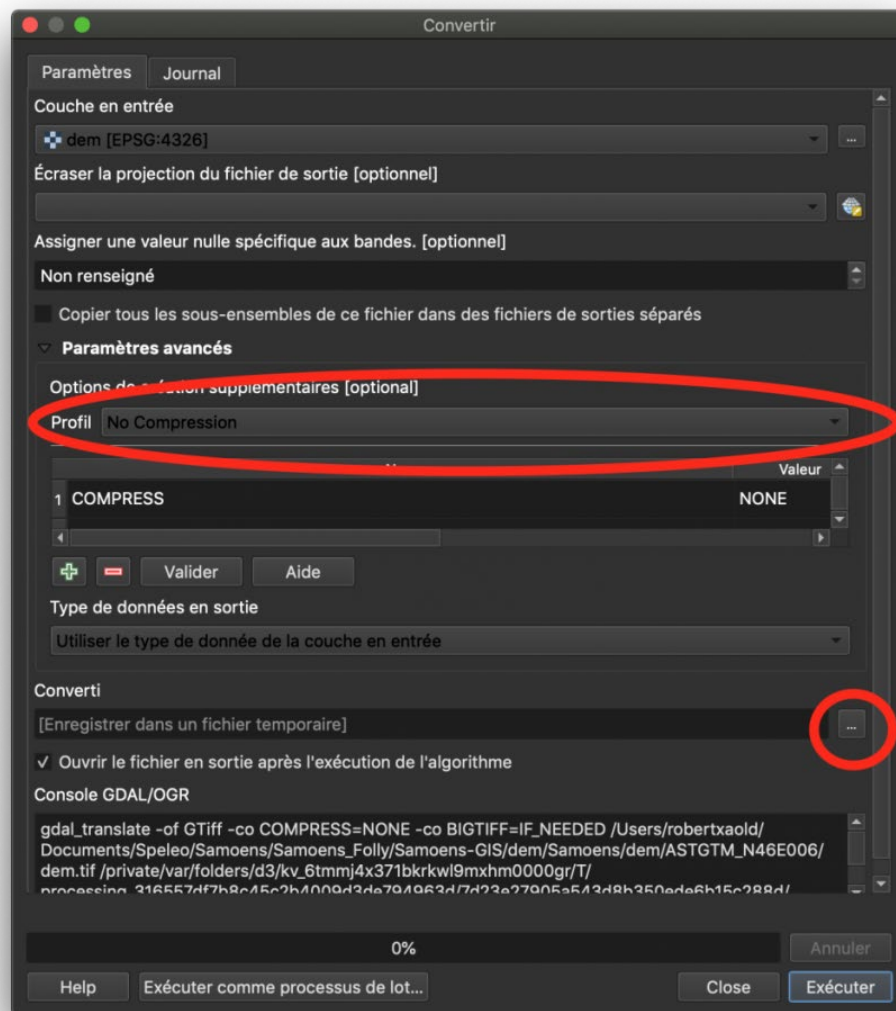


Figure 23 : Conversion d'un Raster dans un autre format.



Dans notre cas, bien penser à ne pas utiliser de compression (grand ovale rouge).

Normalement, cela va produire un fichier .asc. Si vous l'ouvrez avec un éditeur de texte, vous devriez le lire sans problèmes. Les premières lignes donnent :

- le nombre de colonnes (ncols) = nombre de pixels dans la direction x
- le nombre de lignes (nrows) = nombre de pixel dans la direction y
- la coordonnée x du coin inférieur gauche (ll = lower-left) dans la projection utilisée pour ce MNT ; Attention, cette projection n'est pas indiquée...
- la coordonnée y du coin inférieur gauche
- la taille du pixel dans l'unité de projection (si la projection est géographique, l'unité est le degré, si c'est un UTM ou un Lambert, l'unité devrait être du mètre)
- La valeur utilisée pour dire que ce pixel n'a pas de données (NODATA\_value)

Par exemple, en en-tête, nous obtiendrons :

```
ncols 302
nrows 189
xllcorner 324833.651685020013
yllcorner 5105639.090404500253
cellsize 26.514313645002
NODATA_value -9999
```

Si en l'éditant vous avez des signes bizarres, si c'est illisible, c'est que vous n'avez pas mis dans le « Profil » le « No compression »

### **Modifier le MNT pour l'adapter à Therion**

Nous avons notre MNT dans un format texte, formaté de façon à ce que *Therion* le comprenne. Il nous reste à le modifier légèrement pour que *Therion* puisse l'utiliser.

Il faut donc éditer le fichier .asc dans un éditeur de texte. Avant de le modifier, nous pouvons modifier son extension en .th à la place de .asc. Sinon, une fois ouvert dans un éditeur de texte, nous pouvons l'enregistrer sous mnt.th, par exemple.

Pour *Therion* la définition de la surface (MNT) est incluse dans un bloc

```
surface
endsurface
```

Nous devons donc rajouter une première ligne

```
surface
```

puis naviguer à la fin du fichier et rajouter en dernière ligne du fichier

```
endsurface
```

Ensuite, nous l'avons vu, le fichier original est constitué d'un préambule donnant les caractéristiques de la grille MNT, et du tableau de la grille MNT. *Therion* ne le comprend pas, mais nous avons besoin des valeurs

pour calibrer la surface. Il nous faut donc commenter (rajouter le # en début de ligne) toutes les lignes du préambule.

Ensuite, nous allons passer à la calibration de la surface, en rajoutant quelques lignes entre le préambule que nous avons commenté et le tableau de données :

- D'abord, nous devons indiquer la projection du MNT (généralement un UTM ou un Lambert pour la France) avec la commande `cs`, par exemple ici avec l'UTM32N :
  - `cs UTM32N`
- Ensuite, il nous faut indiquer l'unité de la grille. Pour l'UTM ou le Lambert, elle devrait être en mètres, et c'est la valeur par défaut. Nous l'indiquons avec la commande
  - `grid-units m`
- Il nous faut ensuite calibrer la grille, avec la commande
  - `grid <origine x> <origine y> <espacement x> <espacement y> <nombre colonne> <nombre ligne>`
  - où
    - `<origine x>` et `<origine y>` sont les coordonnées (x, y) du coin inférieur gauche,
    - `<espacement x>` et `<espacement y>` sont la taille du pixel en x et en y (le pixel peut ne pas être carré),
    - `<nombre colonne>` et `<nombre ligne>` est le nombre de colonne et le nombre de ligne de la grille, c'est à dire du tableau de données qui suit
  - vous l'avez compris, toutes ces données se trouvent dans le préambule que nous avons commenté, il suffit de faire un copier-coller des bons nombres !
  - Il se peut que la grille ne soit pas au format classique (ça dépend des logiciels utilisés pour produire le MNT), et qu'elle soit en miroir, soit par rapport à la verticale, soit par rapport à l'horizontale. Vous le verrez vite à l'export du 3D ! Pour corriger, il suffit d'utiliser la commande
    - `flip horizontal/vertical`

Nous rajouterons ensuite la commande de drapage de l'image, voir ci-dessous.

Par exemple, notre fichier MNT `mnt.th` ressemblera ainsi à :

```
surface

# Début du préambule original commenté
#NCOLS          44
#NROWS          31
#XLLCORNER      233915.20256810292
#YLLCORNER      9346251.195786513
#CELLSIZE       30.760091552522432
#NODATA_VALUE   -9999
# fin du préambule original commenté

    cs UTM18S # Utilisation de la projection WGS84 - UTM 18 Sud
    bitmap Sat_Bellavista.jpg [0 0 233883 9346185 3508 2479 235299 9347167] #
Insertion de l'image Sat_Bellavista (voir plus loin)
    grid-units meter # Définition de l'unité de la grille
    grid          233915.20256810292          9346251.195786513          30.760091552522432
30.760091552522432 44 31 # Définition de la grille pour Therion

[Notre tableau de données]

Endsurface
```

Et voilà, le MNT est prêt pour être compris et utilisé dans *Therion*.

Pour l'utiliser, il suffit de rajouter, dans le fichier de compilation *.thconfig*, son importation :

```
import mnt.th # Import du MNT pour le 3D
```

### ***Draper une image sur le MNT***

Pour draper une image sur un MNT pour *Therion*, nous devons utiliser un format d'image compris par *Therion*, par exemple le format *.jpg*. Malheureusement, le format GeoTIFF ne convient pas. Il faut donc exporter notre image de drapage en *.jpg*.

Effectuer ses opérations tiennent plus de la bidouille qu'autre chose. En gros, ce qu'il nous faut, c'est :

- produire une image, d'emprise égale ou plus grande à celle du MNT
- projeter l'image dans la même projection que le MNT
- connaître la dimension en pixel de l'image,
- Connaître les coordonnées des coins inférieur-gauche et supérieur-droit de l'image dans le système de projection du MNT

Ce que nous proposons ici fonctionne bien, mais il y a peut-être des manières de faire plus robuste.

### ***Charger une image géoréférencée dans le projet QGIS***

Dans QGIS, charger le projet correspondant à la zone étudiée (le projet construit pour produire le MNT).

Vérifier que la projection du projet corresponde bien à la projection du MNT utilisée.

Pour charger une image, il va falloir utiliser une image géoréférencée. Si vous en avez une sous la main (par exemple, des photos aériennes de l'IGN pour la France, où une photo aérienne que vous avez ortho-rectifiée puis projetée), alors, c'est le moment de la charger dans le projet.

Sinon, avec QGIS, avec un accès à internet, nous pouvons charger des images de fond, qui peuvent être (liste non exhaustive) :

- les cartes Open Street Maps (libre)
- Les cartes et images satellites de Bing
- Les cartes et images satellites de Google
- Et beaucoup d'autres...

Pour pouvoir les charger, dans QGIS, il faut installer le plugin QuickMapService (Pour installer un plugin, aller dans Extension → Installer/Gérer les extensions et chercher l'extension QuickMapService). Une fois l'extension installée, aller sur Internet → QuickMapService → Settings. Aller dans l'onglet « More Services » et cliquer sur « Get Contributed Pack ».

Voilà, maintenant, une fois le projet chargé, vous pouvez choisir un fond dans Internet → QuickMapService

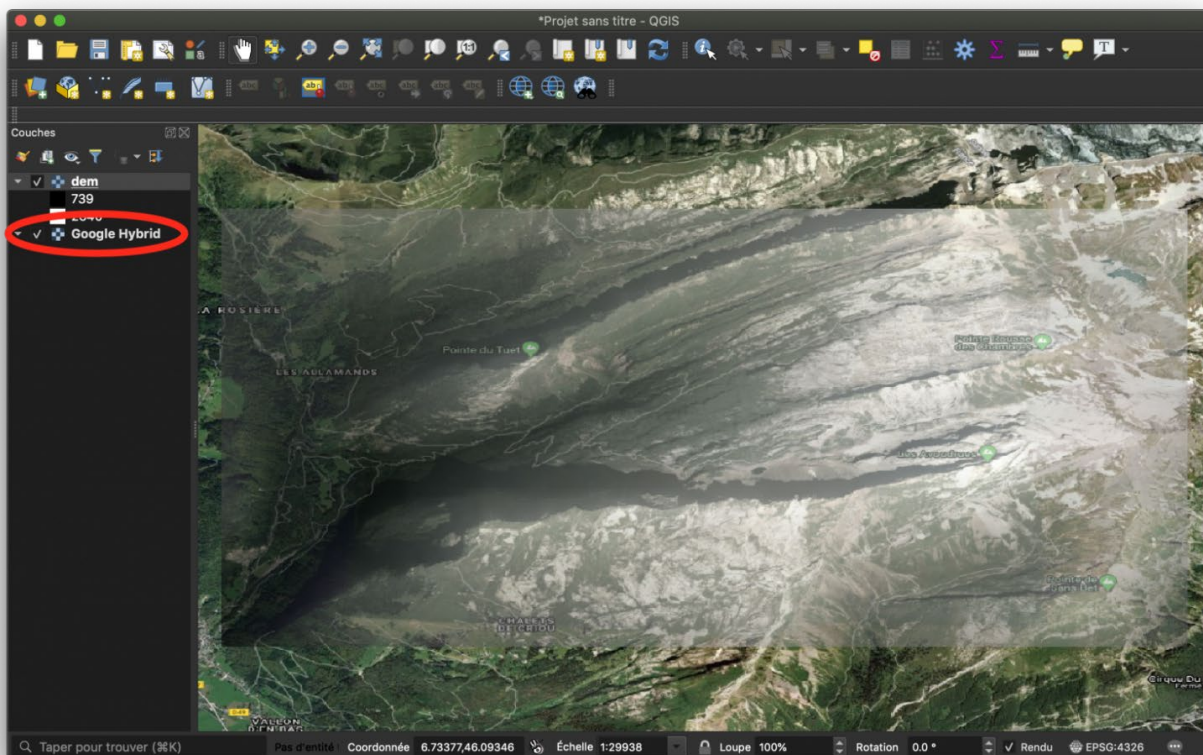


Figure 24 : Insertion d'une image Google hybride comme image de fond, avec l'outil QuickMapService.

### Extraire l'image

Dans les couches de votre projet, vous devriez avoir le MNT exporté pour *Therion*. Faire un clic-droit → zoomer sur la couche. Ainsi, tout l'écran est occupé par ce MNT.

Ensuite, faites remonter l'image de fond que vous voulez utiliser en drapage en premier plan (la faire remonter à la souris dans le panneau des couches).

Passer en mode de mise en page : Projet → Nouvelle Mise en page. Lui donner un nom.

Par défaut, la souris est dans le mode de sélection. La page blanche, c'est la page qui sera imprimée.

Il nous faut passer dans définition du cadre à mettre en page, pour dire où nous voulons notre image sur la page. Pour cela, dans les onglets à gauche, cliquer sur le bouton représentant une page blanche, avec une petite croix rouge dans le coin inférieur droit. C'est le bouton « Ajouter une nouvelle carte à la mise en page ».

Pour faire un cadre dans lequel sera notre image, il suffit de cliquer où nous voulons un premier coin, de laisser appuyé, puis de déplacer le pointeur vers le coin opposé. Vous pouvez faire ceci n'importe où dans la page, mais moins il y aura de blanc autour de l'image, meilleure sera la résolution finale, et plus facile sera le travail de traitement. Faisons donc ce cadre, en essayant d'être le plus au bord possible de la page blanche.

Une fois fait, nous voyons dans ce cadre ce que nous avons dans notre plan de travail, avec tous les objets. Si besoin, vous pouvez jouer sur beaucoup d'option avec le panneau de droite.

Si l'image vous convient, aller dans Mise en page → Exporter au format pdf

Voilà, nous avons une belle image au format pdf, mais le problème, c'est que cette image est bordée par un cadre blanc, qu'il nous faut supprimer.

Ouvrir le pdf avec une application de gestion d'image, comme [Gimp](#) ou Adobe Photoshop. Si vous n'y arrivez pas, avant, exporter votre pdf dans un format image non destructif (.tif par exemple, ou alors, un jpg de grande qualité). Ouvrir cette image là.

Avec le cadre de sélection, sélectionner juste l'image, sans les bords blancs, mais en essayant de ne pas perdre de parties de l'image.

Faites un copier-coller dans une nouvelle image de cette sélection, normalement, vous ne devriez pas perdre en qualité d'image, et vous ne devriez plus avoir de bords blancs.

Enregistrer au format jpg d'excellente qualité. Nous allons utiliser cette image là.

### ***Trouver les données locales et géographiques de l'image***

Nous trouvons les données locales de l'image dans ses propriétés : il faut récupérer le nombre de pixels en x, et de nombre de pixels en y. Les propriétés de l'image peuvent être visualisées soit avec un clic droit sur l'image, ou mieux, avec un éditeur d'image type [Gimp](#) (Libre) ou Adobe Photoshop par exemple.

Il nous faut aussi les coordonnées géographiques de l'image, dans sa projection. C'est plus difficile à obtenir.

Une manière de faire, approximative, est de retourner sur notre projet QGIS, au zoom exact auquel nous avons extrait l'image. En déplaçant le pointeur de la souris sur l'image, nous pouvons lire dans le bandeau inférieur les coordonnées de la localisation du pointeur sur l'image, dans la projection du projet (qui devrait être la même que celle de l'image extraite). Il suffit donc de repérer les pixels de l'image correspondant au coin inférieur gauche et au coin supérieur droit, et de noter leurs coordonnées sur une feuille. C'est approximatif, mais si vous êtes consciencieux, l'erreur de localisation devrait être inférieure aux autres erreurs (GPS, de topographie,...).

### ***Insérer le tout dans le MNT Therion***

Cela ce fait avec la commande

```
bitmap <nom_de_image> <calibration>
```

où :

- <nom\_de\_image> est le nom de l'image que nous voulons draper ; Si l'image est dans un autre dossier, alors, rajouter le chemin devant
- <calibration> permet d'ortho-rectifier l'image, et de la localiser pour pouvoir la coller sur le MNT. Cette calibration peut avoir deux formes :
  - [X1 Y1 x1 y1 X2 Y2 x2 y2 [units]] où :
    - X1 et Y1 sont les coordonnées en pixel de l'image du coin inférieur gauche (0, 0)
    - x1 et y1 sont les coordonnées géographiques réelles du coin inférieur gauche
    - X2 et Y2 sont les coordonnées en pixel de l'image du coin supérieur droit, soit X2 = nombre de pixels dans la direction x et Y2 = nombre de pixels dans la direction y
    - x2 et y2 sont les coordonnées géographiques réelles du coin supérieur droit

- [units] est l'unité des coordonnées vraies, c'est optionnel, si rien n'est indiqué c'est par défaut en mètres, ce qui est valable pour les projections UTM et Lambert.
- En fait, nous pourrions utiliser n'importe quel couple de points sur l'image, mais il est plus facile de connaître les coordonnées de l'image en pixel pour les coins
- [X1 Y1 station1 X2 Y2 station2] où
  - station 1 correspond à une station topographique de référence (par exemple une entrée bien visible sur l'image), que nous pouvons retrouver aux coordonnées X1 et Y1 e(n pixel) sur l'image
  - station 2 correspond à une station topographique de référence (par exemple une autre entrée bien visible sur l'image), que nous pouvons retrouver aux coordonnées X2 et Y2 e(n pixel) sur l'image

Cette commande bitmap doit être insérée dans un bloc « surface – endsurface » (voir plus haut).

### ***L'utiliser dans un layout pour un plan***

Plusieurs commandes, dans un layout dédié à la mise en page d'un plan, permettent de gérer l'image de fond, dans le cas où le fichier de surface est importé.

Pour afficher ou non l'image de fond, il faut utiliser l'option

```
surface <top/bottom/off>
```

avec :

- top signifiant que nous voulons l'image au dessus de tout le reste
- bottom signifiant que nous voulons l'image au dessous de tout le reste
- off signifiant que nous ne voulons pas d'image

Et nous pouvons choisir de rendre la surface plus ou moins transparente avec l'option :

```
surface-opacity 70
```

70 est la valeur par défaut, nous pouvons jouer avec cette valeur, sachant que 100, correspond à l'image originale sans transparence, et que 0 correspond à une image entièrement transparente et donc non visible.

## **10. PARTAGER LA BASE DE DONNEES ?**

Pour la base de données topographiques du massif du Folly, [un article spécifique a été écrit pour l'Echo des Vulcains n°77](#). Cette page en est partiellement tirée.

Construire une base de données topographiques pour la garder uniquement sur son ordinateur, et ne pas la partager est un peu un non-sens : personne ne pourra avoir accès à cette base sans l'accord de l'auteur de la base, ça va poser des problèmes de diffusion, voir des conflits de personnes, et surtout, cela empêche quiconque de participer à l'évolution de la base de données topographiques et interdit aux scientifiques l'accès à ces données qui peuvent être importantes ! C'est donc très égoïste, valeur qui pourtant n'est pas (en tout cas, ne devrait pas être) une valeur de la communauté spéléologique. Il faut donc publier et partager.

Mais publier et partager posent des questions

- Nous verrons dans une autre page comment protéger légalement le travail (voir la page sur les licences).
- Nous pourrions juste permettre le téléchargement de la base de données sur un site internet quelconque (par exemple, celui d'un club de spéléologie), mais cela pose le problème de la gestion des mises à jour, de l'incorporation de nouvelles données par des auteurs différents, et de qui s'occupe de tout coller et mettre à jour.

Une personne pourrait être en charge de gérer l'ensemble de la base de données, de la mettre à jour en fonction des apports des uns et des autres, et aussi de mettre à jour la base de données sur internet. Cela demande beaucoup de travail pour une seule personne.

Un moyen plus efficace et collaboratif serait de mettre en place cette base de données topographique sur une plateforme de travail en ligne dédiée, associée à un logiciel de gestion de versions.

Il existe plusieurs logiciels de gestion de versions (*Subversion – SVN, Mercurial, Bazaar, Git,..*). La plupart sont uniquement des commandes en ligne, ils ne sont pas forcément faciles à prendre en main et l'aide d'une personne ayant des connaissances en informatique pourra aider fortement.

Egalement, il existe plusieurs types de plateformes en ligne (gratuites, libres, payantes,...) disponibles. Une des plateformes conviviales est [Github](#) (gratuite pour l'utilisation que nous pourrions en avoir, mais appartenant à Microsoft).

L'avantage de cette plateforme, c'est qu'elle est associée au logiciel de contrôle de version Git. Ici, cela permet de mettre à jour la base de données en ligne très facilement, de revenir à une version antérieure si besoin, et, pour les non-administrateurs de la base de données, de mettre à jour leur propre version locale de la base de données facilement, et de proposer des modifications aux administrateurs. Enfin, pour cette plateforme, il existe une [interface graphique pour utiliser Git](#) et ses options.

Lever une topographie, dessiner une topographie, gérer une base de données topographiques, c'est un gros travail, demandant énormément de temps, plusieurs centaines, voir milliers d'heures. Généralement, nous ne voulons pas être spoliés de notre travail.

La réponse des spéléos est variée :

- Certains ne disent rien et gardent leur découverte pour eux, et du coup, tout ce qui ont fait est rapidement perdu, et tout doit être à refaire ;
- d'autres ne publient que les images des topographies, parfois sans coordonnées, voir déformées. C'est déjà ça, mais les données sources ne sont pas publiées, nous avons souvent qu'une vision limitée sans intégration dans un système karstique, et puis souvent, les données sources sont perdues...

Pourtant, tout ce travail est d'une importance capitale, a minima pour la recherche scientifique, et pour les futurs spéléos qui prendrons notre relève sur nos chers massifs. Ces bases de données topographiques gagneraient à être publiées. Mais pour que notre important travail ne soit pas spolié, il peut être intéressant de le protéger. Ainsi, nous pouvons le publier.

Pour le protéger, il suffit d'y apposer une licence libre. En effet, libre ne veut pas dire libre de droit. Une licence est régie par un texte de loi... et donc va permettre de protéger le travail.

Aujourd'hui, il existe plusieurs types de licences libres qui dépendent du type d'objet à licencier (code, document,...). Un type de licences bien adapté est l'ensemble de licences [Creative Commons \(CC\)](#). En fonction de vos besoins, plusieurs licences y sont disponibles :

- La licence [CC-BY](#) : c'est une licence d'attribution (by = par), il n'y a aucune contrainte quant à l'utilisation des données, uniquement, l'utilisateur doit citer l'auteur.
- La licence [CC-BY-SA](#) : c'est une licence d'attribution et de distribution dans les mêmes conditions (Share Alike – sa), dans le cas de l'utilisation de la base de données et de sa publication, en plus de citer l'auteur, l'utilisateur doit diffuser la base de données dans les mêmes conditions que celle originale, c'est à dire avec la même licence.
- La licence [CC-BY-ND](#) : Comme pour les deux licences précédentes, l'utilisateur a le droit de copier et diffuser la base de données, même commercialement, mais n'a pas le droit de la modifier.
- La licence [CC-BY-NC-SA](#) : C'est la licence d'attribution (BY) de partage à l'identique (au niveau licence, sa) et non commerciale (NC). L'utilisateur peut faire ce qu'il veut de la base de données, même la rediffuser, mais que s'il cite l'auteur, fournir la même licence, et que ce n'est pas dans un cadre commercial.

Toutes les options peuvent se combiner entre-elles. La base de données topographique du Massif du Folly est sous licence CC-BY-NC-SA.

## 11. QUELQUES EXEMPLES...

Pour l'instant, nous avons développé deux bases de données topographiques :

- La première est la base de données topographique de notre zone d'exploration savoyarde de prédilection, [le massif du Folly](#). Toutes les topographies finales peuvent se trouver sur les [pages dédiées de ce site web](#).
- La seconde est la [base des données topographique du Pérou](#), effectuée avec le concours du GSBM et d'ECA.

Une troisième est en cours de construction. Elle sera sur les cavités explorées à Bornéo, en Indonésie, lors de l'[expédition 2010](#).

Ces deux bases de données sont sous licence CC-BY-NC-SA.



## 12. TABLE DES ILLUSTRATIONS

Figure 1 : Le mode compilation de <i>XTherion</i> avec un <i>thconfig</i> chargé.....	135
Figure 2 : La fenêtre de <i>XTherion</i> après une compilation réussie .....	136
Figure 3 : L'éditeur de dessin après importation d'un fichier <i>xvi</i> . .....	142
Figure 4 : Insertion du premier point Station topographique .....	147
Figure 5 : Dessin de la première ligne « Paroi (wall) ».....	149
Figure 6 : Exemple du dessin d'une entrée, avec les deux lignes de parois ouvertes vers l'extérieur.....	151
Figure 7 : Le dessin d'un pilier.....	152
Figure 8 : Dessin à partir d'un dessin scanné.....	154
Figure 9 : Définition d'un point Section. Noter l'option qui appelle le scrap de la Section.....	161
Figure 10 : Dessin d'une ligne de pente (slope).....	163
Figure 11 : Sélection d'une ligne fermée « libellé (label) » pour créer une aire de type eau (water).....	165
Figure 12 : l'exemple d'une zone complexe avec différents niveaux de superpositions .....	167
Figure 13 : Définition d'un point Section. Noter l'option qui appelle le scrap de la Section.....	170
Figure 14 : Fenêtre de compilation de <i>XTherion</i> avec une erreur de compilation. ....	174
Figure 15 : Exemple de compilation avec des warnings. ....	175
Figure 16 : l'option 'debug on' .....	181
Figure 17 : Exemple de décalages (Réseau de la Combe au Puares, Samoëns, France).....	184
Figure 18 : Quelques panneaux classiques sous QGIS. ....	192
Figure 19 : Outil de reprojection d'un raster (wrap).....	194
Figure 20 : Mode Mise en Page de QGIS avec une carte en cours de travail. ....	195
Figure 21 : Choix du système de projection du MNT.....	198
Figure 22 : Outil de découpe d'un raster sous QGIS, avec choix de la zone de découpe. ....	199
Figure 23 : Conversion d'un Raster dans un autre format.....	200
Figure 24 : Insertion d'une image Google hybride comme image de fond, avec l'outil QuickMapService..	204



# **THERION POUR LES SPÉLÉOS PURS ET DURS**

**(MAIS TOPOGRAPHES QUAND MEME !)**

*Wiki Therion*



# SOMMAIRE

## 1- INTRODUCTION

## 2- COMMENT ÇA MARCHE ?

- 2.1- Généralités
- 2.2- Données LRUD (en français : GDHB pour Gauche, Droite, Haut, Bas)
- 2.3- *Therion* et *XTherion*
- 2.4- Comment l'installer

## 3- LA BALLADE DES DAMES DU TEMPS JADIS

- 3.1- Le plan
- 3.2- Coupe développée
- 3.3- Sections transverses
- 3.4- Le pdf final (enfin !!!)

## 4- LES CHOSES DIFFICILES...

- 4.1- Joindre des lignes d'axe topographique
- 4.2- Joindre des Scraps
- 4.3- Régler l'ajustement (on utilisera layout par la suite !)

---

# 1- INTRODUCTION

*Therion* est la prochaine étape des logiciels de topographie spéléologique. Il fonctionne sous *Windows* (9X, 2000 XP), \*nix et *MacOS X*. Il est sous licence GNU-GPL.

Traditionnellement, un logiciel de topographie spéléologique travaille avec les seules données métriques (depuis, vers, longueur, direction, pente...). A partir de ces données, une polygone de cheminement central (centerline) est calculée, puis cette ligne est exportée vers un logiciel de dessin comme *Illustrator*, où l'on ajoute les parois et autres détails.

*Therion* peut travailler à la fois avec les données métriques et les dessins. Il n'est plus nécessaire d'exporter la ligne d'axe vers un logiciel de dessin. Vous pouvez dessiner une ligne représentant un puits, puis définir son type comme puits et *Therion* dessinera un puits en utilisant les normes de l'UIS ou d'autres groupes spéléologiques. Donc vous n'avez plus besoin de craquer le copyright très restrictif d'*Illustrator* pour y arriver !

*Therion* peut déformer vos dessins pour les adapter aux changements de votre ligne d'axe. Si les données de votre ligne d'axe sont modifiées, habituellement, vous devez redessiner les parois, ... Ce n'est pas utile avec *Therion*. Ses algorithmes tiennent compte de choses comme les superpositions de galeries, etc. Ainsi, vos relevés sont toujours à jour ! Vous n'avez pas besoin de redessiner après avoir réalisé une boucle, corrigé une bourde, ou changer de lot de symboles.

*Therion* peut dessiner des plans sur une seule page, dans la limite de 5 X 5 mètres. Il peut également dessiner des atlas de cavité, avec zone de recouvrement et icône de navigation sur chaque page.

*Therion* peut construire des modèles 3D avec les données LRUD, Ces modèles sont calculés à partir des parois que vous dessinez, ainsi les modèles 3D de *Therion* paraissent normalement meilleurs que les modèles 3D “carrés” calculés par les autres logiciels de topographie souterraine. Les données de surface peuvent être ajoutées.

Enfin, *Therion* peut exporter vos données de ligne d'axe au format SQL, prêtes à être récupérées dans une base de données, vous permettant ainsi de réaliser des recherches complètes sur ces données, ou les intégrer dans un SIG.

Ce guide suppose que vous n'êtes pas un “geek”-spéléologue, c'est-à-dire un spéléo pro de l'informatique, mais un vrai spéléo avec peu, voire pas de connaissances en informatique. Si vous êtes un “geek”-spéléo, ou si vous avez besoin d'utiliser les capacités avancées du logiciel, vous pourrez lire le *Therion Book* (therion.pdf), dans votre répertoire therion/thbook.

Il y a un autre guide pour les non-informaticiens, réalisé par Wookey, baptisé “*Therion - State of the Art Cave-Drawing Software*” (non traduit pour l'instant !), disponible à <http://therion.speleo.sk/download.php>.

## 2- COMMENT ÇA MARCHE...

### 2.1- Généralités

Pour la plupart de vos relevés, vous devrez créer 3 fichiers texte :

- 1) le fichier .th avec vos données topographiques (ligne d'axe en particulier) ;
- 2) le fichier .th2 avec vos dessins ;
- 3) le fichier *thconfig* qui peut contenir les éléments suivants :
  1. liste des éléments que vous souhaitez voir figurer sur votre document final ;
  2. réglages de mise en page, comme le lot de symboles, utilisation de légendes, etc.;
  3. spécification de votre sortie : carte et/ou atlas et/ou modèle 3D et/ou données d'axe au format SQL.

Le plus pratique est d'enregistrer ces 3 fichiers dans le même dossier et de le baptiser du nom de la cavité.

Ainsi, dans *XTherion*, vous pouvez :

1. saisir vos données topographiques dans la fenêtre d'éditeur de texte ;
2. dessiner vos parois et autres détails dans la fenêtre de l'éditeur de carte. Ils seront convertis en commandes et sauvegardés dans un fichier .th2 ;
3. dans la fenêtre du compilateur, régler vos sources, mises en page et export dans un fichier *thconfig*, puis appuyer sur “Compile” pour voir vos instructions transformées en une carte finie, un modèle 3D, ou un fichier .sql.
4. voir votre nouveau modèle 3D dans la fenêtre du visualiseur 3D si vous avez ordonné à *Therion* de le faire.

Les fichiers .pdf finaux peuvent être visualisés avec *Adobe's Acrobat Reader* si vous êtes du côté obscur de la force, *Aperçu* si vous êtes du côté moins obscur, ou *xpdf* si vous travaillez sur un système de type \*nix. Vous pouvez également utiliser *gv*, mais pour les grandes cavités, *gv* est tout simplement beaucoup trop lennnt !!

Il y a des exemples disponibles sur le site de *Therion*. Le mieux est de les utiliser pour faire vos premiers pas avec *Therion* !

## 2.2- Les données LRUD

L'idée d'origine de *Therion* était qu'il n'était pas nécessaire de noter les données LRUD, comme on réalise généralement sous terre des croquis à l'échelle dans les carnets topo. Ces croquis ont des données LRUD implicites, ainsi pouvez-vous les scanner puis les charger dans l'éditeur de cartes comme des "images d'arrière-plan". Ensuite, vous redessinez sur vos croquis.

Depuis la version 0.3.7, *Therion* peut montrer la ligne d'axe avec les données LRUD dans l'éditeur de cartes, ainsi n'avez vous pas besoin d'avoir un scanner ou un croquis à l'échelle. À la place, vous avez le squelette de votre cavité sur lequel vous dessinez vos parois, etc. comme avant.

Vous pouvez choisir la méthode qui vous convient le mieux, voire combiner les deux. Ce guide se concentre sur l'utilisation des données LRUD. Pour savoir comment travailler avec des croquis scannés, lisez le guide de *Wookey* et/ou le *Therion Book*.

## 2.3- *Therion* et *XTherion*

*Therion* est un programme en lignes de commande. Il demeure masqué dans la mesure où nous utilisons *XTherion* comme interface graphique. Basiquement, la tâche de *XTherion* est de convertir les points et lignes que nous dessinons en commandes qui peuvent être traitées par *Therion* quand on appuie sur F9 (n'importe où) ou sur le bouton "Compile" (dans la fenêtre du compilateur).

## 2.4- Comment l'installer ?

Premièrement, télécharger le sur <http://therion.speleo.sk>.

Sur *Windows*, un double-click sur l'icône de l'installateur lancera l'installation. Répondez "Oui" (Yes) et "Suivant" (Next) à chaque question. C'est fait ? Vous pouvez maintenant démarrer *XTherion* en double-cliquant sur l'icône. Comme mentionné précédemment, vous travaillerez principalement avec *XTherion* qui est l'interface graphique de *Therion*.

Sur \*nix, vous pouvez installer à partir d'un paquet binaire si vous êtes sur *Debian* (*Sarge*). `Apt-get` l'installera avec tous les éléments tierces nécessaires...

Si vous n'êtes pas sur *Debian*, vous devrez le compiler. Lisez `thbook.pdf` pour voir comment faire ! Regardez sur la page de Download <http://therion.speleo.sk/download.php> c'est bien expliqué.

# 3- LA BALLADE DES DAMES DU TEMPS JADIS

Pour commencer, nous vous suggérons de faire une topographie d'entraînement, avec une cavité très simple. Aucune des fonctions avancées de *Therion* ne sera expliquée dans le présent chapitre, mais cela suffira pour voir comment saisir des données topographiques, comment dessiner et comment compiler vos cartes.

## 3.1- Le plan

### 3.1.1- Premiers contacts

Donc vous avez démarré *XTherion* et obtenu une fenêtre (noire) de compilateur... Et maintenant ?

Placez votre curseur sur les différents panneaux de contrôle et les boutons et vous voyez ainsi à quoi ils servent (en anglais pour l'instant...). Depuis la version 0.3.7, quand vous placez votre curseur sur un bouton ou un élément de ce type, une bulle d'aide vous explique brièvement sa fonction. La même chose se produit sur la barre des menus, en haut de la fenêtre de *XTherion*. Par la suite, cette barre des menus vous indiquera également des informations sur les objets dans l'éditeur de cartes. Il vous faudra donc y faire attention !

Si vous trouvez ces bulles d'aide désagréables, vous pouvez éditer le fichier *XTherion.ini*. Il est placé dans les sources de *Therion*, dans le dossier "*XTherion*". Ouvrez-le et cherchez la ligne :

```
# set xth(gui,balloons) 1
```

et changez-la en :

```
set xth(gui,balloons) 0
```

La prochaine fois que vous démarrerez *XTherion*, les bulles n'apparaîtront pas (mais les messages continueront de s'afficher dans la barre des menus !).

Evidemment, tous les panneaux de contrôle et les boutons sont désactivés tant que vous n'avez pas ouvert un fichier existant ou que vous n'en avez pas enregistré. De la même façon, certains panneaux de contrôle seront inactifs tant que vous n'aurez pas inséré un élément adéquat. Ainsi, le panneau de contrôle "Points" sera inactif tant que vous n'aurez pas inséré au moins un point. Ne vous inquiétez pas, les messages de la barre des menus apparaissent même si les contrôles sont inactifs.

Nous vous suggérons d'aller à "Fichier→Ouvrir" (2e bouton en partant de la gauche) et de sélectionner un fichier *thconfig* dans les exemples que vous avez chargés. Ensuite, appuyez simplement sur le bouton "Compile" ou F9. Immédiatement à côté du bouton "Compile", vous voyez une case jaune avec la mention "RUNNING". *Therion* est donc en train de créer des choses à partir des commandes se trouvant dans *thconfig*, et les fichiers .th et .th2. Lorsque la compilation est terminée, la boîte de message indique "OK" sur fond vert et vous trouverez dans le dossier de l'exemple choisi, les fichiers .pdf, .thm (modèle 3D) et .sql créés. Ouvrez les fichiers .pdf dans un logiciel de visualisation adapté, les .thm dans la fenêtre de visualisation 3D de *XTherion* et le fichier .sql dans un éditeur de texte...

### 3.1.2- Données d'axe topographique et fichiers .th

Maintenant que vous avez vu de quoi *Therion* est capable, il est temps de faire vos premiers pas. Donc fermer le fichier *thconfig* d'exemple utilisé précédemment et choisissez la fenêtre d'éditeur de texte. Dans cette fenêtre, cliquez sur le deuxième bouton en partant de la gauche et choisissez *template.th*.

Vous voyez que ce fichier contient un grand nombre de lignes débutant par "#". Tout ce qui est compris entre # et la fin de la ligne est du commentaire et est ignoré lors de la compilation par *Therion*... Nous les utilisons pour expliquer ce que chaque ligne utile signifie, quelles options sont disponibles... Donc en dehors des commentaires, *template.th* ne contient qu'une douzaine de lignes de commande réelles. Vous avez sans doute juste besoin de lire attentivement ce fichier et de faire quelques modifications comme suggéré dans les commentaires. Mais nous indiquons ici un guide pas à pas plus détaillé.

### 3.1.2.1- Style et ordre des données.

Le squelette d'un fichier *.th* peut être résumé de la façon suivante :

```
-----  
survey FOO -title "FOO Cave"  
  
centerline  
data normal from to length compass clino  
  
(données mesurées ici)  
  
Endcenterline  
  
endsurvey  
-----
```

Tout ce qui se trouve entre les 2 éléments d'une paire *centerline* / *endcenterline* appartient au même levé topographique (*survey*). Un "survey" peut contenir un grand nombre de choses, et en particulier d'autres surveys. Donc chaque survey doit avoir un nom particulier pour permettre une structure hiérarchique (voir plus loin pour le détail...). Pour l'instant, la chose intéressante est la *centerline* (ou ligne d'axe topographique)... Elle peut également être baptisée *centreline*. Comme vous le savez, tout le monde n'est pas naturellement anglophone !

A l'intérieur d'une paire *centerline* / *endcenterline*, il y a :

- des lignes de données générales : unités, déclinaison, ordre des données mesurées... ;
- des lignes de relevés topographiques proprement dites.

Dans cet exemple; il n'y a qu'une ligne de données générales : celle d'ordre et de style des données. Elle précise le style et l'ordre de vos données topographiques. Vous pouvez le changer selon vos besoins.

"normal" signifie que vous utilisez le style normal en faisant votre relevé, soit : boussole, clinomètre et décamètre. Si vous utilisez du topofil, remplacer "normal" par "topofil". Si vous êtes plongeur, modifier "normal" en "diving". Il y a encore d'autres styles... Pour les connaître, consulter le *thbook.pdf*.

Les données relevées doivent suivre le même ordre que les mots-clés de la ligne "data". Si vous notez vos données selon un autre ordre (par exemple, vous notez la longueur en dernier), changer simplement l'ordre des mots-clés. Dans ce cas, la ligne deviendrait :

```
data normal from to compass clino lenght
```

Ou, si vous notez sous forme LRUD, vous utiliserez :



data normal from to lenght compass clino left right up down

Il y a des mots-clés pour tous les besoins... Par exemple, si vous mesurez en direct et inverse, vous pouvez utiliser les mots-clés “backcompass” et “backclino”. Ou si vous êtes plongeur, vous pouvez utiliser “depth” (profondeur). La ligne d'ordre des mots-clés peut contenir autant de mots-clés que nécessaire !. Pour la liste complète des mots-clés, reportez-vous au... thbook.pdf.

### 3.1.2.2- Unités

Par défaut, les unités de longueur sont le mètre et celles d'angle les degrés (360° dans un tour !). Si vous utilisez un appareil en grades (400) et disons que vous mesurez vos longueurs en yards (si vous utilisez cette dernière unité, avez-vous réellement besoin de cette version française ??), votre fichier .th devrait se présenter ainsi :

```
-----  
survey FOO -title "FOO Cave"  
centerline  
units lenght yard  
units compass clino grad  
data normal from to length compass clino  
  
(données mesurées ici)  
  
endcenterline  
endsurvey  
-----
```

### 3.1.2.3- Date, déclinaison et le reste...

Vous pouvez ajouter d'autres lignes de données générales à vos données d'axe topographique. En fait, vous pouvez spécifier (au minimum) la date, la déclinaison, les participants au relevé, et l'ordre et le style des données. Ajouter les lignes d'unité si vous n'utilisez pas des degrés et des mètres. Les topographes de précision peuvent indiquer sd (standard déviation) et une ligne de calibration pour les réglages fins de leurs instruments. Encore une fois, pour la liste complète et la description de toutes les lignes disponibles, reportez-vous au thbook.pdf (je m'attaquerai à sa traduction dans un certain temps...).

Si vous avez plus d'une séance de levé, il serait mieux de préciser la déclinaison pour le survey, et pas pour le centerline. Cela permet un calcul automatique de la déclinaison. Plus de détail plus loin.

Voici un fichier .th typique d'un spéléo pur et dur. C'est le fichier template.th que vous trouverez dans le dossier “Exemples”, sans commentaires. Vous pouvez éditer les parties FOO, etc. pour l'adapter à vos besoins. Noter qu'il est préparé pour une boussole en grades (400) et un clino en degrés (360). Les données LRUD sont utilisées.

```
-----  
survey FOO -title "FOO Cave"  
centerline  
date YYYY.MM.DD  
declination -3 degrees  
team "Firstname Lastname"  
-----
```

```
team "Firstname Lastname" # add as many team lines as you need
units compass grad
data normal from to length compass clino left right up down
```

(données mesurées ici)

```
endcenterline
endsurvey
```

---

### 3.1.2.4- Données relevées (enfin !)

Ecrivez-les simplement en texte tabulé (c'est-à-dire chaque donnée du même point séparée par une tabulation). Une fois que vous avez la ligne de style et ordre des données, vous pouvez cliquer sur le bouton "**Scan data format**" dans le panneau de contrôle "**Table data**". L'ordre des boîtes de texte sera modifié selon votre ligne d'ordre des données. Si votre ligne d'ordre des données a plus de mots-clés que de boîtes de dialogue dans le panneau "Table data", les boîtes manquantes seront ajoutées. Ainsi, vous pouvez saisir vos données d'une manière plus visuelle. Mais tant que vous n'aurez pas appuyé sur "Tab" pour passer d'une fenêtre à l'autre, vous n'aurez sauvegardé aucune saisie...

Ainsi, vos données auront probablement le format classique, disons :

```
0      1      16.25  345      -10      1.00  0.85  6      1.60
1      2      4.10   275      1        0.30  0.90  2      1.60
```

Je suis presque certain que vous avez déjà vu ce format de données auparavant 😊

### 3.1.2.5- Points fixes

Pour donner des coordonnées géographiques générales à votre ligne d'axe, vous devez disposer d'au moins un point fixe (mesuré au GPS par exemple). Il s'agit d'une station avec des coordonnées réelles, que vous ajouterez à vos données relevées sous la forme de la ligne suivante :

```
fix      0      575628 4476124      1250
```

Comme vous le voyez, ce point fixe est la station 0 avec des coordonnées UTM. Vous pouvez avoir autant de points fixes que nécessaire. Plus il y en a, mieux c'est ! Par habitude, je les note au début de mes données relevées. Ainsi, mon fichier .th ressemble à ça :

---

```
survey irurixo -title "Irurixo Koba"
centerline
date 2005.02.26
declination -3 degrees
team "Mac Theknife"
team "Juxe Euskalduna" # add as many team lines as you need
units compass grad
data normal from to length compass clino left right up down
```

```
fix      0      575628 4476124      1250
fix      42     575100 4476655 1310
```

```
0      1      16.25  345    -10    1.00   0.85   6      1.60
1      2      4.10   275     1     0.30   0.90   2      1.60
(puis la suite des données relevées)
```

```
endcenterline
endsurvey
```

---

Bien. Maintenant, nous pouvons créer notre premier fichier *thconfig*, nous permettant ainsi d'exporter un xvi (image vectoriel *XTherion*) avec notre ligne d'axe topographique. Cette image sera chargée dans l'éditeur de cartes comme "image de fond".

### 3.1.3- Thconfig

Dans le menu "fenêtre", sélectionner "Compilateur" (ou F3) puis dans fichier, choisissez "Ouvrir" (ou dans la barre des menus, cliquer sur l'icône correspondante) et choisissez *thconfig*. Encore une fois, il s'agit d'un fichier avec commentaire. Le squelette en est le suivant :

---

```
source template.th
layout xvi-export
scale 1 1000
grid-size 1 1 1 m
endlayout
export map -fmt xvi -layout xvi-export -o irurixo.xvi
```

---

Il y a 3 parties dans un fichier *thconfig* :

a) source *template.th* est la partie listant les éléments à utiliser. Donc nous précisons ici les fichiers qui seront compilés par *Therion*.

b) les spécifications de mise en page. Tout ce qui se trouve à l'intérieur de la ligne *layout / endlayout* fait partie de la mise en page. Ces mises en page doivent être dotées de nom pour préciser quelle mise en page appliquer à quel export... On peut avoir autant de mise en page que l'on en a besoin (ou que l'on veut !).

"*scale 1 1000*" règle l'échelle de sortie au 1/1000e.

"*grid-size 1 1 1 m*" crée une grille avec un pas 1m dans toutes les directions (X, Y et Z).

Si vous travaillez sur de grandes cavités, ce réglage peut provoquer des problèmes de mémoire lors de la compilation et il est donc préférable de régler le pas de la grille à 10 10 10 m. Encore une fois, vous pouvez spécifier d'autres unités que le mètre.

c) les spécifications de la sortie. Pour l'instant, nous n'avons qu'une sortie. Nous allons exporter une carte (une seule page, pas un atlas). Cette carte sera au format xvi (option "*-fmt xvi*"). Cette carte utilisera la mise en page *xvi-export*, préalablement définie (option "*-layout xvi-export*"). Enfin, l'option "*-o irurixo.xvi*" indique le nom du fichier xvi qui doit être créé.

C'est bon ? Alors appuyez sur F9 ou le bouton "Compile" et attendez que le dialogue à côté du bouton "Compile" indique "OK" sur fond vert. Votre nouveau fichier xvi doit se trouver dans le dossier d'exemple. Si vous avez le message "ERROR" sur fond rouge, vous pouvez voir dans la partie inférieure du compilateur ce qui est arrivé. Cette partie inférieure affiche le fichier log (c'est-à-dire les différents événements se produisant lors de la compilation) et habituellement la cause de l'erreur se remarque alors... Vous pouvez corriger les lignes fausses puis réappuyer sur le bouton "Compile" (ou F9).

Vous êtes maintenant prêt à dessiner sur votre ligne d'axe...

### 3.1.4- Données dessinées et fichier .th2

#### 3.1.4.1- Images d'arrière-plan

Choisissez l'éditeur de cartes dans le menu "fenêtre" (F2). Puis choisissez Nouveau dans le menu Fichier. Le fond noir doit apparaître jaune maintenant. Il est temps de donner un nom et de sauver le nouveau fichier .th2 pour que le chemin d'accès aux autres éléments (comme les images d'arrière-plan) puisse être enregistré.

Le panneau de contrôle "images d'arrière-plan" se trouve en bas à droite de la fenêtre de l'éditeur de cartes. Cliquer sur le bouton "Insert". Si vous n'avez pas encore enregistré votre fichier .th2, un message d'alerte vous demandera de le faire. Cliquer sur "OK" et une fenêtre d'enregistrement apparaîtra.

Après que le nouveau fichier .th2 ait été sauvé, une autre fenêtre apparaîtra, vous permettant de choisir l'image d'arrière-plan à charger. Choisissez celle que vous venez de charger (se terminant par .xvi) Ne vous souciez pas des autres boutons : ils sont utiles pour les croquis scannés.

Vous pouvez régler le zoom dans le panneau "Drawing area", juste au dessus du panneau "Background images". Cliquer sur le rectangle juste à côté de "zoom 100%" pour voir le menu de zoom. Vous pouvez également faire défiler la zone de travail.

Vous êtes maintenant prêt à réaliser votre premier scrap !

#### 3.1.4.2- Scraps...

Les dessins sont divisés en "scraps". Tout ce qui se trouve entre les deux termes d'une paire "*scrap/endscrap*" appartient au même scrap. Nous y trouvons des points, des lignes, des surfaces. Vous pouvez voir les scraps et ce qu'ils contiennent dans le panneau "File commands" (le plus haut).

Pour les petites cavités simples, tout le dessin peut être contenu dans un scrap. Mais, si vous avez à faire à des cavités plus grandes, la réalisation d'un seul scrap peut vous conduire à des problèmes de manque de mémoire lors de la compilation. Ce n'est pas le cas de notre exemple. De la même façon, si vous avez des galeries qui enjambent des zones sous-jacentes, la seule façon de faire est de diviser en plusieurs scraps, car ces derniers ne peuvent pas avoir d'intersection entre eux (plus de détails plus loin...). Pour l'instant, on veut juste apprendre à dessiner, donc nous ne ferons qu'un seul scrap.

Dans le panneau "File commands", cliquer sur "Insert scrap". Le panneau "Scrap control" s'ouvrira automatiquement et *XTherion* vous suggérera de le baptiser "scrap1". Pour l'instant, c'est très bien. Ensuite choisissez une projection pour ce scrap. Comme nous dessinons le plan, choisissez "plan" dans le menu. Vous pouvez cliquer sur "Update" pour être sûr que *XTherion* a bien pris en compte les modifications, mais normalement, il le fait tout seul. Le scrap est maintenant créé et vous voyez une paire de lignes dans la liste "File commands".

#### 3.1.4.3- Les points

Retournez dans le panneau "File commands". Cliquer sur le menu déroulant à côté d'action pour voir les options d'action. Choisissez "Insert point". Le bouton "Insert scrap" se transformera en "Insert point" se changera en "Insert point". Cliquez dessus pour commencer à insérer des points. Toutes ces étapes peuvent être faites par Ctrl-P. Nous sommes désormais en mode "insertion de point". En bas de la fenêtre, vous voyez un rectangle rouge indiquant ce que nous faisons : insert point.

Par défaut, *XTherion* créera des points avec le dernier type utilisé. Le plus rapide est donc de créer tous les points du même type (par exemple, les stations), puis de changer le type (stalactite par ex.) pour insérer toutes les stalactites, et ainsi de suite.

#### **3.1.4.3.1- Les points de stations**

En premier lieu, nous devons placer et référencer les stations. L'image .xvi montre la ligne d'axe avec les stations. Cliquer sur l'une d'elles. Le panneau de contrôle des points s'ouvre et *Therion* réglera le type sur "station". Dans la boîte option, vous verrez "-name X", où X est le nom de la station ainsi qu'elle est nommée dans les données de la ligne d'axe (fichier .th). Elle a été référencée automatiquement car l'image xvi contient les données nécessaires.

Donc, lorsque vous travaillez sur des images xvi, vous n'avez pas besoin de vous soucier du référencement des stations dessinées. Ce ne sera pas le cas si vous travaillez sur des croquis scannés.

Vous pouvez quitter le mode "insertion des points" en appuyant sur la touche (esc), ou en cliquant 2 fois sur le même point. La boîte rouge en bas de la fenêtre apparaît de couleur verte et affiche le message "select object".

Maintenant, vous pouvez cliquer sur les points pour les sélectionner. Lorsqu'ils sont sélectionnés, vous pouvez les déplacer par glissement.

Pour supprimer un point, sélectionnez-le, allez dans le panneau "File commands", cliquer sur le menu "Action", choisissez "Delete", puis appuyez sur ce qui est désormais le bouton "Delete" (au-dessus du menu déroulant "Action"). Ou sélectionner le point et appuyer simplement sur "Ctrl-D" (ou la corbeille, dans les icônes en haut à droite).

Chaque scrap a besoin d'au moins 2 points de référence aux stations de ligne d'axe. A partir de ces deux points, la mise à l'échelle et la rotation peuvent être calculées dans la mesure où nous avons la position des points sur l'écran (en pixels) et leurs coordonnées réelles (à partir des données de ligne d'axe). Par habitude, j'indique toutes les stations, et pas seulement deux. Cela augmente la précision.

S'il n'y a pas au moins deux stations référencées dans le scrap, il doit être mis à l'échelle manuellement. C'est le cas lorsque vous sélectionnez la projection "none" (pour les sections transverses en particulier), donc ne vous en souciez pas pour l'instant !

#### **3.1.4.3.2- Les flèches de pente**

Donc vous avez référencé toutes vos stations dans le scrap. Et maintenant ?

Et bien, vous pouvez placer des flèches indiquant la pente. Entrez dans le mode d'insertion des points, cliquer pour insérer un nouveau point, puis aller dans le panneau de contrôle "point" et choisissez "gradient" dans le menu "type" (cliquer sur le triangle, alors que la valeur apparente dans cette boîte n'est pas sélectionnée).

Les flèches de pente doivent pointer dans une direction. Pour ce faire, cocher le bouton "orientation". Maintenant, sur votre dessin, le point montrera une flèche. Cliquer sur la pointe de la flèche et glissez-là

selon la direction désirée. On peut faire la même chose en tapant un angle dans le dialogue à côté du bouton "orientation" (360°). Maintenant, vous pouvez cliquer ailleurs pour insérer une autre flèche et ainsi de suite...

Nous le savons, c'est toujours un point, et pas une flèche de pente. *XTherion* n'est pas un logiciel WYSIWYG. Vous ne voyez aucun symbole à l'écran, que des lignes et des points. Ne vous inquiétez pas : après la compilation, les symboles seront affichés dans les cartes .pdf terminées. Et vous pouvez placer le curseur sur les objets et voir ce qu'ils sont dans la barre d'état.

#### **3.1.4.3.3- Les étiquettes**

Maintenant, vous pouvez insérer des étiquettes, comme "P100", ou "entrée". Insérez un point où l'étiquette doit se trouver, puis allez dans le panneau de contrôle "Point", sélectionnez "label" dans le menu "type". Ensuite, dans la boîte de dialogue "options", écrivez : -text "Ceci sera affiché"

Comme vous pouvez le deviner, après compilation, la carte pdf terminée affichera "Ceci sera affiché" centré au-dessus de la position du point correspondant.

Il y a quelques astuces à connaître... Si vous écrivez ceci : -text "<it> Ceci sera affiché" .... la carte compilée affichera "Ceci sera affiché" en italique.

Ou encore : si vous écrivez : -text "Ceci sera <br> affiché" .... la carte compilée affichera "Ceci sera affiché" sur deux lignes.

Ou encore : si vous écrivez : -text "<right>Ceci sera <br> affiché" ... la carte compilée affichera le texte sur deux lignes, aligné à droite. Les attributs <right>, <left> et <center> n'ont d'effet que si il y a également un attribut <br>.

#### **3.1.4.3.4- Les sections**

Cette option est utilisée pour associer les sections transverses. Chaque section transversale doit être dans son propre scrap, avec la projection "none". Plus de détails par la suite.... Insérez un point là où une section transversale a été relevée, choisissez le type "section dans le menu, et dans la boîte de dialogue d'options, écrivez : scrap <nomduscrap>

En faisant cela, vous référencez un autre scrap où la section transversale est dessinée.

#### **3.1.4.3.5- Autres types de points**

Comme vous pouvez le voir dans la liste du type de point, il y en a d'autres que ceux que nous avons mentionnés ici. La plupart sont des symboles pour les spéléothèmes, les remplissages de galerie, les fins de galerie, l'équipement... Nous ne les détaillerons pas ici car il s'agit seulement d'une introduction. La plupart de ces points s'utilisent sans option particulière (par exemple tous les spéléothèmes). Nous vous suggérons d'essayer de les utiliser et si après compilation ils n'apparaissent pas comme vous le souhaitez, reportez-vous au thbook.pdf (pages 18 à 20) pour voir les options mentionnées.

#### **3.1.4.4- Les lignes**

Normalement, les lignes des parois sont les premières lignes que l'on dessine dans l'éditeur de cartes. Cependant, nous avons décrit les points en premier car il nous apparaissait plus simple de commencer par les objets les plus simples qui sont en l'occurrence les points. Les lignes sont constituées de points. Vous pouvez dessiner les lignes avant les points, cela n'a aucune importance.

Pour dessiner une ligne, appuyer sur “Ctrl-L” ou classiquement, par le menu “file commands”, choisissez “Insert line” dans le menu Action et cliquer sur ce qui est devenu le bouton “Insert line”. Encore une fois, vous verrez la boîte rouge en bas de la fenêtre avec cette fois la mention “insert line point”.

#### **3.1.4.4.1- Dessiner des lignes droites**

Comme mentionné précédemment, les lignes sont composées de points. Maintenant que nous sommes dans le mode d'insertion de points de ligne, cliquer sur la zone de dessin, puis cliquer sur la même zone à une autre endroit. Vous devez obtenir une ligne droite entre ces deux points... Vous pouvez poursuivre votre ligne en cliquant où vous avez besoin. Ainsi, vous aurez une ligne constituée de segments droits entre les différents points.

Pour finir une ligne (c'est-à-dire sortir du mode “insertion de point de ligne”), vous pouvez soit a) appuyer sur la touche Esc, b) re cliquer sur le dernier point que vous venez de faire. La boîte rouge en bas de la fenêtre est devenue verte, et vous êtes revenu sur le mode de sélection d'objet. Et vous pouvez appuyer sur “Ctrl-L” pour ajouter une autre ligne, ou cliquer sur des objets pour les sélectionner et les éditer. Pour supprimer une ligne entière et tous ces points, sélectionnez-la et appuyer sur “Ctrl-D”, ou allez dans le menu “File commands”, choisissez “Delete dans le menu Action et appuyer sur ce qui est devenu le bouton “Delete”.

Lorsque vous cliquez sur une ligne, un de ses points est automatiquement sélectionné. Vous le remarquerez par le cercle rouge qui l'entoure. Normalement, vous cliquez directement sur les points que vous souhaitez modifier. Si vous cliquez sur un segment de la ligne, ce sera le point suivant en direction de la fin de ligne qui sera sélectionné.

Lorsqu'un point de ligne est sélectionné, vous pouvez le déplacer par un simple glissement.

Pour supprimer seulement un point sur la ligne (et non pas toute la ligne), sélectionnez-le, allez dans le menu “Line control”, cliquez sur le rectangle à côté du mot “Edit” pour voir le menu et choisissez “Delete point”.

Pour insérer un point dans la ligne, vous devez sélectionner le point se situant APRES la zone (en allant vers la fin de la ligne) où vous voulez insérer votre point, aller dans le menu “Line control”, cliquer sur le menu déroulant et choisir “Insert point”. Maintenant vous pouvez cliquer sur la ligne et insérer votre point à l'endroit souhaité. Lorsque vous avez terminé, appuyer sur la touche “Esc” ou re cliquer sur le point que vous avez sélectionné au début pour sortir de ce mode.

Vous pouvez recouper une ligne. Sélectionner le point où la coupure doit être faite, puis aller dans le menu “Line control”, cliquer sur le menu déroulant et choisir “Split line”. Si à l'endroit où vous souhaitez couper la ligne il n'y a pas de point, vous devez en insérer un au préalable.

Il est temps de mentionner le taquet jaune affiché au début de chaque ligne sélectionnée. Si vous regardez du premier point vers le point suivant, ce taquet se trouvera toujours sur le côté gauche. C'est important car il indique le côté où seront dessinées beaucoup de choses. Par exemple, les hachures pour les ressauts seront dessinées de ce côté. Lorsque vous dessinez des parois, vous devez les dessiner dans le sens inverse des aiguilles d'une montre (c'est de plus en plus difficile de savoir à quel sens ça correspond avec les montres à affichage digital !), ainsi le taquet jaune se trouvera à l'intérieur et non à l'extérieur. Si vous n'avez pas tracé votre ligne dans le bon sens, ne vous inquiétez pas ! Allez dans le menu “Line Control” et cochez la case “Invert”. Cela inversera l'ordre des points et le premier point deviendra le dernier et le taquet jaune s'affichera sur le côté opposé.

Vous pouvez dessiner une aire fermée. Pour ce faire, terminez simplement sur le premier point de votre ligne, votre aire sera automatiquement fermée et vous reviendrez automatiquement dans le mode de sélection. Ou allez dans le menu "Line control" et cochez "Close".

#### **3.1.4.4.2- Dessiner des courbes**

Comme nous venons de le voir, pour dessiner une ligne droite, il suffit de cliquer là où l'on souhaite implanter un point. Pour réaliser une courbe, vous devez en plus glisser. C'est-à-dire : choisissez un endroit approprié pour votre point, cliquez et maintenez votre bouton de souris enfoncé et déplacez la souris : vous obtenez une courbe. Plus vous déplacez la souris et plus la courbe est prononcée. Lorsque votre courbe vous convient, relâchez le bouton de votre souris. Et voilà... Et vous continuez de la même façon pour les autres points...

Attention, le dernier point dessiné est toujours sélectionné et porte 2 petits carrés rouges. Ce sont des poignées. Elles contrôlent la courbe. La première contrôle la courbe avant le point. Vous pouvez la glisser et voir comment cela modifie la courbe. La seconde contrôle la courbe en aval du point. Pour l'instant, il n'y a pas de courbes après notre point, donc elle paraît inutile, mais dès lors qu'un point existe en aval, elle prend tout son sens.

Il y a une troisième poignée, magenta, sur le point précédent. C'est ainsi qu'apparaîtra la seconde dès lors que vous aurez créé le point suivant. Actuellement, vous pouvez la déplacer pour prononcer plus ou moins la courbe mais sans changer sa direction : elle est fixe.

Dans la mesure où deux points de ligne consécutifs influencent le segment intermédiaire, les transitions seront toujours douces. Et si vous voulez, après un segment arrondi, un segment droit ? Et bien, vous tirez le second point rouge jusqu'à ce qu'il soit sur le point de ligne. Cela annule son influence et donc aucun arrondi ne se fera. Mais il y a une meilleure méthode : dans le panneau de contrôle "Line point control", vous verrez trois boutons rouges à cocher (ils sont cochés) : «(previous) , smooth , » (next).

Décocher "»". La deuxième poignée n'est maintenant plus visible. Remarquer également que le bouton "smooth" est décoché automatiquement. Ainsi, la prochaine transition ne sera pas adoucie ! Vous pouvez également décocher "«" et ainsi les deux transitions ne seront pas adoucies. Ca ne vous convient pas ? Recochez "smooth" et vous aurez les deux transitions adoucies. Cocher "«" et "»" seulement si vous voulez préciser l'adoucissement.

Encore une fois, vous pouvez éditer les points après que la courbe ait été faite. Vous pouvez :

1. cliquer sur n'importe quel point pour le sélectionner, et le glisser à sa nouvelle position ;
2. déplacer les poignées pour avoir des courbes plus ou moins prononcées. Si vous ne voyez pas les poignées, c'est sans doute parce que les boutons "«", "smooth" et "»" sont décochés. Donc si vous en avez besoin, cocher les ! Vous aurez ainsi 2 poignées rouges sur le point sélectionné que vous pouvez tirer dans n'importe quelle direction indépendamment et 2 poignées magenta sur le point suivant et le point précédent qui ne peuvent être déplacées que dans une direction.

Vous n'avez pas besoin de créer un grand nombre de points. Avec un peu de pratique, il est facile de dessiner n'importe quoi en plaçant des points uniquement au centre et au point d'inflexion des courbes. Pour la qualité de votre tracé, moins il y a de points, mieux c'est !

Maintenant que vous pouvez faire une ligne de la forme que vous souhaitez, voyons les types de ligne de *Therion*...

#### **3.1.4.4.3- Les types et sous-types de ligne**



Le type se choisit dans le panneau "Line control". Cliquer sur le triangle à côté de "type" pour voir le menu. Choisissez celui qui vous convient. Dessinez vos lignes. Souvenez-vous que *XTherion* créera par défaut les lignes avec le dernier type utilisé. Donc le mieux est de dessiner toutes les lignes du même type les unes après les autres (par exemple les parois), puis de passer au type suivant, etc.

D'autres options s'appliquent aux lignes. Ce sont les "subtype options" (sous-types). Par exemple pour un mur: "préssumé", "non topographié", "glace", etc. Les sous-types disponibles pour chaque type de ligne sont décrits plus bas. Il faut savoir qu'il y a deux moyens d'attribuer un sous-type à une ligne:

- *La ligne entière doit être affectée:*

Ecrire dans la case *Options* de la fenêtre *Lignes* du panneau de contrôle "-subtype #". Ne pas écrire les guillemets et remplacer le # par le sous type choisi, par exemple *presumed*

- *Une portion de la ligne doit être affectée:*

Une paroi n'a pas besoin d'être divisé en petites parties selon le sous-type. Vous pouvez utiliser une ligne de type paroi et affecter les sous-types appropriés aux points de la ligne à partir du début de la ligne (où se trouve la petite coche jaune). Le sous-type "bedrock" ramènera la ligne au type par défaut (qui est "bedrock"). Il faut modifier le sous-type du point de la ligne dans la petite fenêtre noire de la partie Contrôle du point de ligne du panneau de contrôle. Ecrire simplement *subtype xxxx* (sans tiret). Cela change le sous-type du paroi à partir de ce point, jusqu'à la fin de la ligne ou que le sous-type d'un autre point soit changé.

#### 3.1.4.4.3.1- Les parois

C'est le type par défaut proposé par *XTherion* car vous avez probablement déjà une paroi. Ce sont les limites des scraps. La plupart des symboles sont masqués s'ils sont hors de ces limites. C'est pratique, car ainsi vous n'avez pas besoin d'ajuster les autres lignes exactement. Ainsi, une fois que vos parois sont dessinées, vos contours peuvent s'étendre au-delà de ces parois. Les parties hors-limites ne seront pas visibles et pas imprimées. Cependant, comme les aires sont limitées par les lignes qui les recoupent, il est mieux que vos lignes se terminent exactement sur la paroi. Plus de détail plus loin...

La plupart des symboles sont donc masqués par les limites de scrap que constituent les parois. Pour désactiver cette fonction, précisez dans la boîte de dialogue "options : -clip off

Dans ce cas, ce symbole ne sera pas masqué. Vous pouvez également préciser "-clip on" pour qu'il soit coupé. Cependant, certains symboles ne seront jamais masqués (en particulier les étiquettes).

Maintenant, un cas particulier... Vous avez une galerie qui se sépare en deux, puis après quelques mètres, les deux branches se réunissent. Les parois internes peuvent être vues comme une sorte de "pilier". Mais c'est une source de confusion pour *Therion* car il y a maintenant 4 parois... Qu'est-ce qui est hors des limites, et qu'est-ce qui n'y est pas ? Dans ce cas, sélectionner la paroi intérieure (le "pilier") et précisez dans la boîte options : -outline in

Ainsi, *Therion* saura quels sont les parois externes...

Sous-types de parois: *presumed* (supposé), *underlying* (sous-jacent), *unsurveyed* (non topographié), *sand* (sable), *clay* (argile), *pebble* (cailloux), *debris* (débris), *blocks* (blocs), *ice* (glace). Pour chaque cas, vous obtiendrez le symbole correspondant dans le lot de symboles que vous avez choisi (UIS, ASF, SKBB...).

#### 3.1.4.4.3.2- Autres types de lignes

La plupart sont très explicites et n'appellent aucune option. Comme vous pouvez le devinez, `contour` permet de dessiner des courbes de niveau, `pit` dessine des puits, etc.

`Rock-border` et `rock-edge` sont utilisés pour dessiner respectivement les limites et les arêtes internes des blocs.

`Section` est utilisé pour indiquer l'emplacement d'une section transversale. Nous reviendrons en détail sur ce type dans la description des sections transverses.

`Slope` permet de dessiner un contour avec une coche pointant la direction de la pente. Pour ce symbole, vous devez cocher "L-size" dans le panneau "Line point control".

`Border` est utile lorsque vous devez dessiner une ligne fermée qui définit une aire. Dans ce cas, vous souhaitez sans doute qu'elle soit invisible. Pour cela, dans la boîte "options", précisez : `-subtype invisible`

Et cette limite sera invisible. Ce sous-type s'applique à la plupart des lignes, pas uniquement aux limites. Pour plus de détails sur les différents types de lignes et les options qui s'y rapportent, voyez les pages 21 à 23 du `thbook.pdf`.

### 3.1.4.5- Les surfaces

En général, on utilise les surfaces (`areas`) pour préciser certains remplissages, tels que l'eau, les siphons, l'argile, le sable... Les surfaces sont limitées par des lignes. Ces lignes peuvent être de n'importe quel type, mais elles doivent être closes ou définir un espace clos (en recoupant d'autres lignes). Pour être sûr que ces lignes se recoupent réellement, il est préférable de faire dépasser les ligne des parois. Elles seront de toute façon masquées pour les parties dépassant. On peut obtenir le même effet en utilisant le type "border" avec l'option `"-subtype invisible"`.

Souvenez-vous que *XTherion* utilise le dernier type choisi comme valeur par défaut pour créer la prochaine surface. Donc le plus efficace est de définir d'abord toutes les surfaces d'un type, puis de changer de type, etc. Comme pour les autres objets, vous trouverez un menu "type" dans le panneau "Area control".

Pour insérer une nouvelle surface, appuyer sur `Ctrl-A` ou par le processus classique, dans le panneau "File commands", choisissez "Insert area" dans le menu "Action", puis cliquer sur ce qui est désormais le bouton "Insert Area". Le panneau "Area control" s'ouvrira automatiquement et vous serez alors dans le mode "insert area border" (ajout d'une limite de surface), ainsi qu'indiqué dans la boîte rouge, en bas à droite. Maintenant, cliquer sur les différentes lignes constituant les limites de votre surface. Vous devez tourner dans le sens des aiguilles d'une montre ou dans le sens inverse. C'est-à-dire que vous devez cliquer successivement sur les lignes qui se recoupent. Si vous cliquez sur deux lignes qui ne se recoupent pas, vous obtiendrez un résultat inattendu...

Remarquez que lorsque vous cliquez sur une ligne, cela apparaîtra ainsi dans le panneau "Area control" : 12-268-340. Ce sont les noms des lignes (ID) affectés automatiquement par *XTherion*. Lorsque nous avons créé ces lignes, elles n'avaient pas reçu de nom et cela ne posait aucun problème. Mais maintenant que nous souhaitons qu'elles soient utilisées comme limites, elles ont besoin d'un nom. Si vous allez dans le panneau "File commands control", vous verrez que ces noms figurent pour les lignes sélectionnées. Vous pouvez rebaptiser une ligne avec un nom plus explicite en allant dans le panneau "Line control" et en éditant l'ID dans la boîte "ID".

Revenons au panneau "Area control". Vous pouvez cliquer sur un nom dans la liste et cliquer sur le bouton "Delete" pour le retirer de la liste. Les lignes ainsi traitées ne sont pas supprimées, elles ne sont simplement plus considérées comme des limites.

Lorsque toutes les limites ont été définies, appuyer sur la touche "Esc". Vous reviendrez dans le mode de sélection des objets et vous pourrez de nouveau appuyer sur Ctrl-A pour définir une nouvelle surface. Notez qu'il n'est pas possible de sélectionner une surface sur le dessin... Vous devez aller dans le panneau "File commands control" et cliquer sur la ligne appropriée pour sélectionner la surface correspondante. Une des limites de cette surface sera sélectionnée et vous verrez ainsi si c'est bien la surface que vous souhaitez sélectionner.

Si vous ne vous en êtes pas encore rendu compte, c'est la même chose pour les lignes et les points. Si vous cliquez sur une ligne dans la liste "File commands", cet objet sera sélectionné sur le dessin. Réciproquement, si vous sélectionnez un objet sur le dessin, sa ligne sera surlignée dans la liste. Les listes des autres panneaux fonctionnent de la même manière.

Pour supprimer une surface complète, sélectionner-la dans la liste "File commands", puis appuyer sur Ctrl-D ou choisissez "Delete" dans le menu "Action" et cliquer sur le bouton "Delete".

Pour une liste complète des types de surface et des options correspondantes, reportez-vous au thbook.pdf, pages 23 et 24.

## 3.2- Coupe développée

Pour faire une coupe développée, vous travaillez presque comme vous le faites pour le plan. C'est-à-dire que vous exportez une image xvi avec la ligne de cheminement étendue, puis vous dessinez les points, les lignes et les surfaces.

Donc vous devez :

1. étendre votre ligne d'axe ;
2. éditer le fichier *thconfig* pour exporter l'image xvi avec la ligne d'axe étendue ;
3. dessiner ainsi que vous l'avez fait pour le plan.

### 3.2.1- Etendre la ligne d'axe

Depuis la version 0.3.7, *Therion* étendra par défauts tous les points vers la droite. Pour modifier cela, vous pouvez éditer vos données de ligne d'axe pour ajouter certains "extend".

Vous pouvez ajouter "-extend left" à la ligne "centerline" (dans le fichier .th) et tous les points seront développés vers la gauche.

Si vous avez besoin de développer quelques stations à droite, indiquer "extend right" juste avant le premier, puis indiquer "extend left" juste après le dernier. C'est tout !!

Oui, c'est très simple... Vous pouvez également indiquer, en lieu et place de "right" et "left", "normal" et "reverse".

Dans l'exemple suivant, nous développons toutes les stations vers la gauche, mais les stations 34 à 40 doivent être étendues vers la droite. A la fin, nous avons deux petites visées correspondant à de petites prolongations mais que nous ne souhaitons pas voir sur l'image xvi. Voici le fichier :

---

```
survey irurixo -title "Irurixo Koba"  
centerline -extend left # <<== MAIN CHANGE. ALL POINTS WILL EXTEND TO LEFT NOW
```

```

date 2005.02.26
declination -3 degrees
team "Mac Theknife"
team "Juxe Euskalduna" # add as many team lines as you need
units compass grad
data normal from to length compass clino left right up down

fix 0 575628 4476124 1250
fix 42 575100 4476655 1310
0 1 16.25 345 -10 1.00 0.85 6 1.60
1 2 4.10 275 1 0.30 0.90 2 1.60
(more survey data)
31 32 2.80 50.00 29.00 0.40 0.50 0.40 0.40
32 33 4.40 116.00 3.00 3.00 6.00 1.40 0.80
extend right
34 35 8.50 20.00 -5.00 1.00 2.00 1.25 0.25
35 36 3.70 45.00 4.00 0.80 2.00 0.25 0.25
36 37 3.60 51.00 13.00 0.25 0.25 0.25 0.15
37 38 2.00 41.00 16.00 0.00 2.50 2.00 0.20
38 39 4.60 115.00 20.00 1.50 1.00 2.00 1.00
39 40 4.00 393.00 -32.00 1.20 2.00 1.00 0.80
40 2 3.80 133.00 15.00 0.30 0.50 1.20 0.80
extend ignore
41 10 3.10 159.00 6.00 1.00 2.50 1.20 0.80
11 42 4.00 137.00 0.00 0.20 0.80 2.50 1.00

endcenterline
endsurvey

```

---

### 3.2.2- Exporter la ligne d'axe développée en xvi

Vous devez ajouter la ligne suivante à votre fichier de config : `export -proj extended -fmt xvi -o grotte_developpee.xvi`

Dans le fichier d'exemple, décommentez simplement la ligne.

Maintenant, vous pouvez appuyer sur <F9> ou cliquer sur le bouton "Compile" pour créer l'image correspondante. Maintenant, vous travaillez exactement comme vous l'avez fait pour le plan.

### 3.2.3- Dessiner la coupe développée

Bien sûr les fichiers .th2 peuvent contenir autant de scraps que vous voulez... La projection du scrap n'importe pas. Ouvrez simplement votre fichier .th2 et insérez la nouvelle image d'arrière-plan. Vous pouvez la déplacer par glissement (double clique-droit et glisser) ou plus précisément en indiquant les coordonnées en pixels du coin supérieur droit de l'image insérée, puis en cliquant sur le bouton "Move to".

Dessinez vos lignes, référez vos stations, etc. Comme vous le voyez, tout ceci fonctionne exactement comme pour le plan.

## 3.3- Sections transverses

Bien qu'une section transverse puisse théoriquement comprendre 2 stations voire plus, dans la pratique, cela ne se produit pas. Au mieux, il y aura une seule station si la section a été placée sur une station. La plupart du temps, il n'y aura même pas de station sur la section. Or *Xtherion* a besoin de 2 stations au moins pour mettre le scrap à l'échelle automatiquement.

Dans ce cas, nous devons donc mettre à l'échelle manuellement. Après avoir dessiné et mis à l'échelle, nous devons placer notre section sur les positions appropriées des scraps de plan et de coupe, et dessiner une ligne indiquant la position et la direction (vers le fond ou vers l'entrée), toujours sur le plan et la coupe. Nous allons voir tout cela en détail...

### **3.3.1- Dessiner les sections transverses**

Ouvrez votre fichier .th2 et insérez un nouveau scrap avec la projection "none". Ensuite, appuyez sur Ctrl-L et dessinez-la... Vous pouvez la placer où vous voulez sur le dessin, mais il est plus pratique de la dessiner sur une grille. Il sera ainsi plus facile de dessiner à l'échelle puis de régler l'échelle.

### **3.3.2- Mise à l'échelle manuelle**

Lorsque vous ouvrez un nouveau fichier .th2, il y a 2 carrés rouges aux angles inférieurs du dessin. Ces points sont utilisés pour la mise à l'échelle manuelle. Après avoir dessiné le plan et la coupe développée, ils ne sont plus visibles. Si l'objet sélectionné sur le dessin n'est pas un scrap mis à l'échelle manuellement, ils ne seront pas visibles du tout. Dans le panneau "File commands control", vous devez aller sur la ligne du scrap de votre section que vous venez de créer. Ils doivent maintenant être visibles. Cependant, si vous avez un peu trop zoomé, vous ne pouvez pas voir l'ensemble de votre dessin et donc vous devez vous déplacer sur ce dessin, ou zoomer à 25%. Une fois que vous les avez repérés, vous pouvez les déplacer.

Disons que votre section mesure à peu près 5 mètres de haut pour 2 mètres de largeur. Placer le premier point sur un point d'intersection de la grille proche du coin inférieur gauche de votre section. Placer le second sur une intersection proche de l'angle supérieur droit. Aller maintenant dans le panneau "Scrap control", boîte de dialogue "real scale points". Régler les coordonnées du premier point à 0,0 et celles du second à 2,5. Régler l'unité si besoin est (par défaut, ce sont des mètres). Et voilà...

La taille à laquelle vous avez dessiné votre section transversale importe peu : la taille finale sera le résultat des valeurs que vous aurez précisées pour les points "real scale". Donc si vous souhaitez dessiner des sections transverses plus grandes que l'échelle du plan ou de la coupe développée, vous devez donner des coordonnées fausses : ainsi, dans le cas précédent, si vous dessinez le plan et la coupe au 1/1000e et que vous voulez que les sections apparaissent au 1/500e, vous indiquerez les coordonnées 0,0-4,10 au lieu de 0,0-2,5.

### **3.3.3- Positionner les sections transverses**

Les sections transverses s'utilisent conjointement avec les plans et les coupes donc vous devez avoir les points correspondants dans les scraps de plan et/ou de coupe. Il est facile d'ajouter des points de section dans le scrap de la section transversale lui-même. Dans ce cas, la section ne sera pas visible dans le pdf final.

Dans le panneau "File commands control", choisissez le scrap de plan et/ou de coupe approprié et cliquez sur une ligne lui appartenant. Insérez un point et sélectionnez le type "section". *Xtherion* ajoutera les nouveaux objets SUR la ligne sélectionnée, donc ne cliquez pas sur la ligne du scrap elle-même, sans quoi votre point ne serait pas dans le scrap désiré... Malgré tout, ne vous faites pas de souci si votre point fut inséré hors du scrap : cliquez sur la ligne du point en question et déplacez-le avec les boutons "Move up",

Move down“ et “Move to” jusqu'à ce qu'il soit à l'emplacement désiré. Faites également attention à ce que la section est centrée sur ce point dans le pdf final.... Donc placer le comme il faut !!

Lorsque le point est créé dans le bon scrap, vous devez indiquer dans la boîte d'options : -scrap <scrapname> où scrapname est le nom du scrap de la section transverse. Comme vous le devinez, le point de section appelle le scrap de section transverse lors de la compilation. Vous pouvez placer autant de points de sections appelant le même scrap que vous le souhaitez (habituellement, un pour le plan, l'autre pour la coupe).

### 3.3.4- Dessiner les lignes indiquant les sections

Les lignes de section sont utilisées pour :

1. indiquer le point où la section transverse a été positionnée ;
2. indiquer la direction dans laquelle elle a été relevée (vers l'entrée ou vers le fond). Ces lignes peuvent être en L, traversant ainsi la galerie. Une longue ligne joint la section et l'endroit où elle a été positionnée, mais elle est invisible sur la galerie. Au-delà de la galerie, une petite ligne indique, à l'aide d'une flèche, la direction dans laquelle la section transverse a été prise. Pour obtenir cela, il y a quelques astuces...

Nous pouvons penser que nous devons tracer une ligne droite, mais nous sommes dans l'erreur !! En fait, nous devons dessiner une courbe, de façon à ce qu'il y ait des points de contrôle permettant le réglage des dimensions de la ligne pour la sortir de la galerie.

Disons que nous avons un point de section sur une galerie. Cliquer sous ce point et tirer jusqu'à atteindre la paroi. Ensuite, aller sur le côté opposé, cliquer et glisser jusqu'à atteindre la première paroi. En glissant jusqu'aux parois, vous placez les points de contrôle à proximité immédiate des parois et chaque morceau de la ligne sera dessiné correctement sur le pdf. Si vous avez tiré vos courbes à l'intérieur de la galerie, les morceaux de la ligne indiquant la section seront visibles DANS la galerie.

Même si vous avez tiré une courbe, vous aurez une ligne droite entre les deux points et les morceaux de ligne seront aussi longs que vous aurez fait glisser la souris.

Pour avoir une flèche, vous devez ajouter l'option suivante (si vous avez relevé vers le fond !!): -direction end

Cela signifie qu'une flèche sera apposée sur le point final. Comme expliqué précédemment au sujet des lignes, la flèche sera dessinée sur le côté gauche de la ligne, mis en évidence par le tiret jaune du premier point. Donc pour obtenir une ligne qui pointe vers la gauche, vous devez dessiner le premier point en dessous du second et utiliser l'option "-direction begin", et la flèche sera portée sur le premier point. Pour obtenir une section qui pointe vers la droite, le premier point doit être au-dessus du second et vous devez utiliser l'option "-direction end".

## 3.4- Le pdf final (enfin !!!)

Pour obtenir un fichier pdf, il faut simplement ajouter une ligne de la forme suivante dans le fichier de configuration : export map -o map.pdf -layout map\_layout Le fichier créé sera un pdf selon la mise en page définie préalablement dans "map\_layout"....

Plus de détails ultérieurement !

À suivre...



# LE MANUEL DE THERION

*Wiki Therion*







# 1 INSTALLATION DE THERION

Ce chapitre décrit l'installation de *Therion* et le lancement d'un exemple fournit avec le programme.

Une fois terminée, vous aurez un logiciel *Therion* installé, et vous testé un exemple de ce que *Therion* peut faire.

## 1.1 Téléchargement

---

*Therion* a besoin de plusieurs programmes externe (c.à.d., des programmes qui existent indépendamment de *Therion*) : *cavern*, *Tex*, *METAPOST*, *pdfTeX* et *XTherion* qui ont également besoin de *Tcl/Tk*. Tous ces programmes sont disponibles gratuitement sous *Linux*, *Windows* et *Mac OS X*.

Vous pouvez télécharger *Therion* à partir du site web : <http://therion.speleo.sk/download.php>.

### Linux

Les utilisateurs de *Linux* (spécialement ceux de la version *Debian*) n'auront aucun problème, car ces programmes (excepté pour *cavern* qui est une partie de la suite "*Survex*") sont généralement inclus avec leur distrib *linux*, et ainsi, déjà installés. Si un programme est manquant, il suffit de l'installer avec le gestionnaire de packages de la distrib, ou en dernier ressort, de récupérer la source du programme à partir du site web, dézipper le fichier, configurer le *makefile*, le compiler et l'installer. De plus sous *Debian* (ou *Ubuntu*), il vous suffira de taper dans la console la ligne de commande

```
apt-get install therion therion therion-viewer therion-doc
```

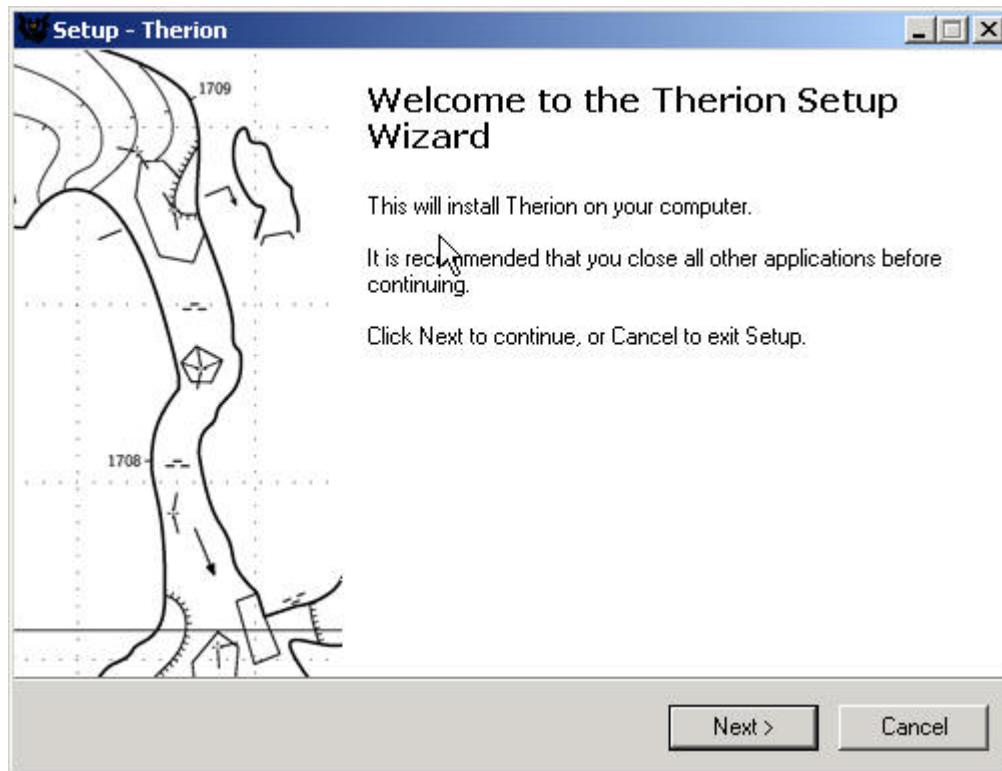
pour que tous les packages dont vous avez besoin soient installés en cascade sans que vous ayez à vous en inquiéter. Merci *Linux*.

Mais à ce jour (janvier 2008) seul le package 'therion' peut être chargé de cette façon. Cela permet tout de même de mettre à jour la quasi totalité des dépendances avant de faire l'installation à la main (voir le site web <http://packages.debian.org/testing/science/therion>) des packages *therion-viewer* *therion-doc* et *therion*.

### Windows

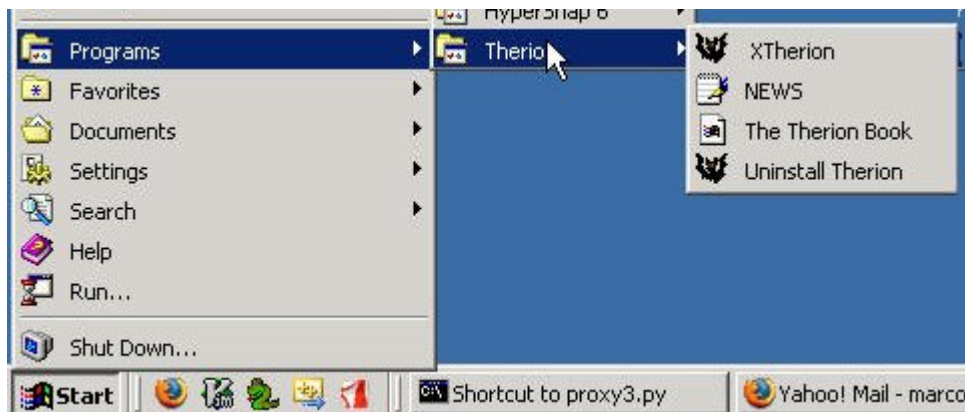
La version de *Therion* pour *Windows* peut être téléchargée en version exécutable. Vous n'avez besoin que de lancer le programme d'installation "therioT-full-0.5.1.exe" (le numéro de la version peut changer).

Quand vous exécutez le fichier d'installation, vous avez une fenêtre d'installation à l'écran comme celle ci-dessous. Vous devez indiquer le répertoire d'installation, et le nom du groupe de démarrage ("Démarrer | Programme") qui sera utilisé par *Therion* (vous pouvez accepter la proposition par défaut).



L'installation de *Therion*, installe *Therion*, *XTherion* et les logiciels nécessaires, *cavern*, METAPOST, Tex, pdfTeX, et tout ce qui est nécessaire.

Quand l'installation est terminée, vous avez le sous-menu "*Therion*" dans "Démarrer | Programme". Sélectionnez le sous-menu "*XTherion*" pour démarrer *XTherion*.



## Mac (par M. Sluka)

Il n'y a pas de programme précompilé pour Mac. Vous devrez :

1. Installez l'outil de développement, *XCode* à partir du kit d'installation Mac OS X
2. Installez *Survex*: téléchargez le fichier source tgz à partir du site web, détarez-le, ensuite configure, make, et installez-le. (Utilisez toujours sudo au lieu de su).
3. Installez le package *TeX* utilisez l'installateur ii à partir de <http://ii2.sourceforge.net/tex-index.html>

4. Installez *AquaTcl/Tk* à partir de <http://tcltkagua.sourceforge.net/> Il y a un *Tcl/Tk* intégré à l'installation de Mac OS X, mais l'installateur BI installera la dernière version.
5. Pour les derniers éléments, vous aurez besoin également de *ImageMagic*. Vous pouvez télécharger les binaire sur <http://www.imagemagick.org/download/binaries/ImageMagick-universal-apple-darwin8.7.0.tar.gz>
6. Téléchargez le package *Therion*. Suivez les instructions dans l'annexe du *The Therion Book*. Pour l'installation, utilisez `sudo make install`.
7. Vous pouvez installer le visualisateur PDF *xpdf*. Il est meilleur qu'*Acrobat Reader* car il ne verrouille pas les fichiers ouverts, *Therion* pourra alors écraser les fichiers lors des recompilations. Avec *acoread* vous devez systématiquement fermer les cartes PDF avant de relancer une compilation. Vous pouvez trouver *xpdf* pour Mac OS X at ... Vous aurez besoin d'installer également le *DarwinPorts*: <http://darwinports.com>. Si port ne marche pas, exécutez la commande `export PATH=$PATH:/opt/local/bin` à partir du terminal. Installez alors *xpdf*: `sudo port install xpdf`.

Sous Mac OS X il y a plusieurs problèmes. Vous devez utiliser les “control key” au lieu des “command key” normales, le bouton droit de la souris ne marche pas, etc.

---

*Le reste de cette section décrit l'installation des programmes utilisés par Therion à partir des sources. Il n'est pas nécessaire de connaître cela pour utiliser Therion, et vous pouvez passer à la section suivante (si vous n'en avez pas besoin).*

### 1.1.1 Cavern

Cavern est une partie de *Survex*, une suite de programmes pour calculer les données topographiques. *Survex* est distribué sous licence GPL et peut être téléchargé librement sur <http://www.survex.org>. Les sources du programme sont disponibles, pour ceux qui veulent le compiler. Il est également disponible sous forme de binaires pour Linux (fichiers deb et rpm), et Windows ou Mac OS X.

### 1.1.2 Tcl/Tk

*XTherion* est écrit en Tcl/Tk. C'est pourquoi vous avez besoin de l'interpréteur Tcl/Tk pour l'exécuter. Si vous n'avez pas le module *BWidget*, vous pouvez l'ajouter : téléchargez le fichier *BWidget-1.7.0.tar.gz*.

NOTE: Je ne me rappelle plus si l'on a besoin également de *tclunit.tcl*.

Si vous prévoyez d'utiliser des images scannées au format PNG or JPEG, vous aurez également besoin des modules d'extensions *tkimg*: *tkimg1.3.tar.bz2*.

TeX

*METAPOST* (program)

(program)

### 1.1.3 TeX, METAPOST et pdfTeX

TeX est généralement inclus dans la distribution Linux et contient également METAPOST. Si ce n'est pas le cas, vous pouvez télécharger TeX et METAPOST à partir de CTAN (Comprehensive TeX Archive Network) ou à partir de TUG (TeX users group).

Je voudrais dire un mot sur les problèmes possible qui peuvent arriver, et comment les résoudre. Malheureusement (ou par chance), je n'ai jamais eu de problèmes. Si quelqu'un a une expérience sur ce point ...

pdfTeX est une version de TeX qui produit une sortie en PDF, a partir du DVI.

#### 1.1.4 *Therion*

*Therion* peut être téléchargé a partir de <http://therion.speleo.sk/download.php>.

Le fichier compressé (therion-0.5.1.tar.gz) contient

- les fichiers sources du programme *Therion*;
- l'éditeur *XTherion* des fichiers de données *Therion*. Il est construit en Tcl/Tk;
- la documentation: le répertoire *thbook* et les pages de manuel *Therion* et *XTherion*;
- des exemples, dans le répertoire *tests*;
- fichiers de données (définitions), dans le répertoire *lib*;
- les fichiers METAPOST, dans le répertoire *mpost*;
- un visualisateur 3D en OpenGL ( *thtom*);
- le répertoire *tex* avec les fichiers *therion.tex*, contenant les commandes TeX de *Therion*;
- glyfs et l'encodage dans le répertoire *texenc*;
- le répertoire *thchencdata* avec l'encodage des données;
- le répertoire *thlang* pour la localisation (langues locales);
- le répertoire *extern* contenant des outils pour *Therion*;
- le répertoire *thcdt* avec des fonctions pour la triangulation de Delaunay, utilisées par la reconstruction 3D des modèles.

Après avoir decompress2 les archives, vous devez les compiler avec la commande "make". Par défaut, "make" génère le programme "*Therion*" et "*XTherion*" et la documentation (le *Therion* book). Si tout va bien, vous pouvez installer *Therion* avec la commande "make install". Si vous n'installez pas *Therion*, vous ne pourrez jamais utiliser *Therion* et *XTherion* a partir du répertoire ou vous avez décompressé le fichier d'archive.

Le Makefile est configure pour Linux. Ceux pour Windows et Max OS X sont commentes. Si vous devez utiliser l'un d'eux, vous devez configurer le compilateur en tapant respectivement "make config-win32" ou "make config-macosx". Cela suppose aussi que vous avez un environnement de développement "make", [MinGW](#) ou [gcc](#). Bien sûr, la solution la plus simple est de récupérer une version compilée du programme.

Pour installer thtom vous devez aller dans le répertoire *Therion/thtom/linux* et exécuter "make". Une fois finit, copier le répertoire Tom0.2 dans le répertoire lib de votre distribution Tcl/Tk (dans mon cas c'était /usr/lib/tcllib1.6). Ce répertoire doit contenir la librairie dynamique libtom.so et le fichier pkgIndex.tcl.

## 1.2 Passer le logiciel en Français

Une fois l'installation réalisée, pensez à éditer le fichier `therion.ini` présent dans le répertoire d'installation pour modifier la langue ( ajouter la ligne "langue fr" )

```
### Output character encodings ###
# encoding-default ASCII
# encoding-sql ASCII

### Default regional settings ###
# language en_UK
language fr
# units metric
```

Pour que les légendes des topographies soient en Français.

De la même façon, éditez le fichier `XTherion.ini` pour modifier la langue utilisée dans l'interface du logiciel :

```
## preferred language for messages
::msgcat::mlocale fr
```

Voir l'annexe 1.A de ce chapitre pour plus d'infos.

## 1.3 Exécution

---

*Therion* est un logiciel pour des topographies de haute qualité. Au cœur de ce système, il y a le langage *Therion*, c'est un langage pour définir via des fichiers texte les éléments graphiques de la topographie, c.à.d. des points, des lignes, des symboles, et ainsi de suite.

Les fichiers texte, écrits dans le langage *Therion*, sont utilisés par le programme "*Therion*" pour générer la topographie en PDF. Ce programme se base sur d'autres programmes pour calculer les données topographiques, "cavern", et pour produire des dessins de qualité, METAPOST, TeX, et pdfTeX [2](#) .

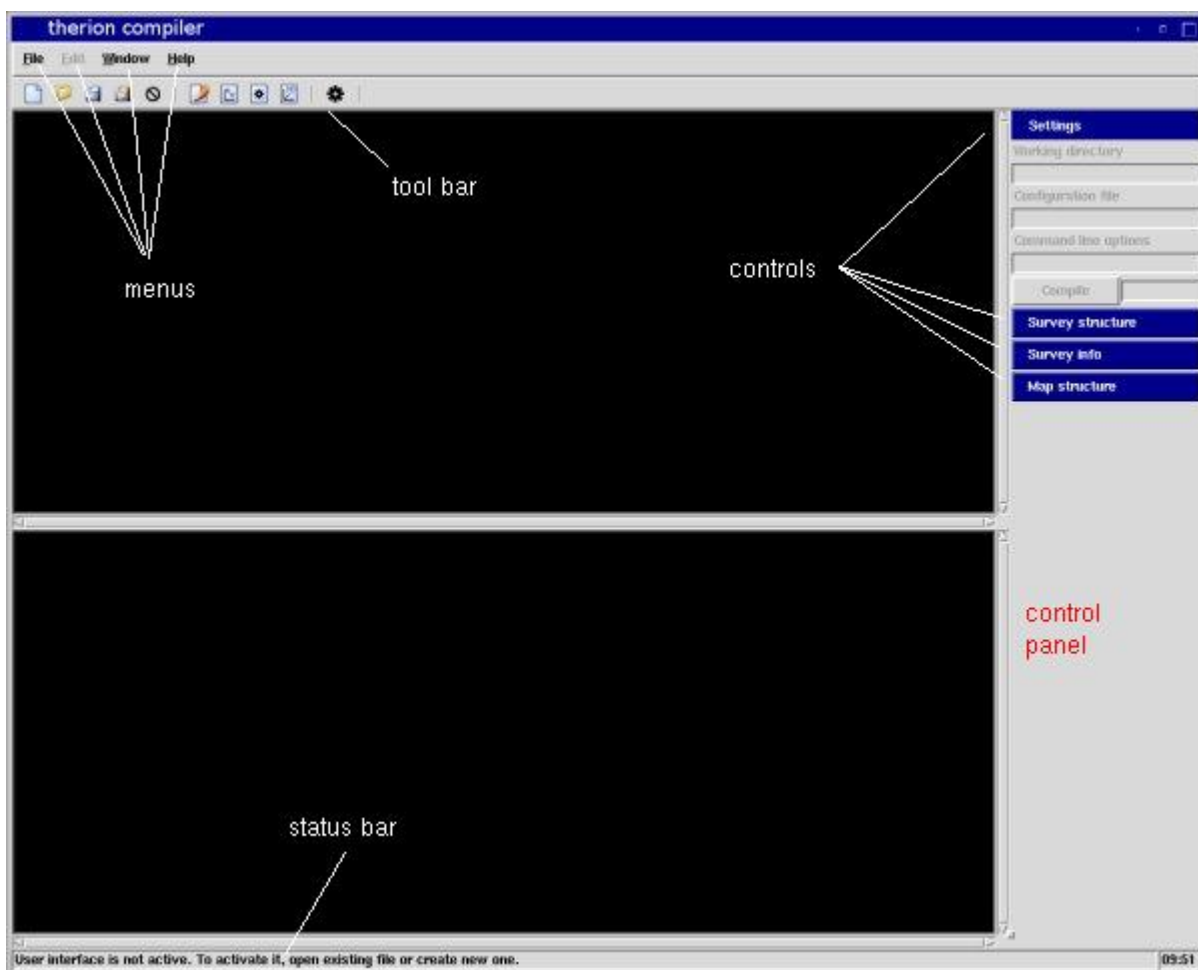
Alors qu'il est facile d'écrire et de modifier les fichiers de données topographiques avec un simple éditeur de texte, faire de même pour les fichiers de dessin l'est beaucoup moins, même si cela reste possible. Il est préférable d'utiliser un programme avec une interface graphique. Un tel programme, "*XTherion*", est distribué avec *Therion*. N'importe quel autre programme graphique compatible avec le langage */Therion/* remplirait également la tâche. "*XTherion*" est écrit en Tcl/tk, c'est pourquoi il est nécessaire d'avoir l'interpréteur Tcl/Tk installé sur son poste.

*XTherion* n'est pas un programme WYSIWYG (ce que vous voyez est ce que vous aurez), ce qui rend l'utilisation moins conviviale. D'un autre côté, l'approche suivie par *Therion* qui sépare les données de leur présentation (et/ou du résultat final), donne plus de liberté pour les résultats (impressions), et rend la maintenance plus simple. Par exemple, la fermeture d'une boucle, est plus facilement prise en compte lorsque les données utilisées pour générer la carte sont toujours les données originales (et non des données obtenues par résultat d'un calcul).

Assurez-vous que les répertoires contenant le programme *Therion* (et les autres programmes utilisés) sont bien inclus dans le "PATH". Les utilisateurs sous Windows qui ont installés *Therion* avec le programme d'installation, ne devraient pas avoir de problème.

Les utilisateurs sous Linux ne devraient pas avoir de problème non plus. Pour vérifier que tous les programmes sont accessibles dans le PATH tapez la commande “which cavern mpost tex pdftex *XTherion*”. Si le programme les trouve, le chemin complet est affiché. Sinon, le nom du programme, avec un point-virgule est affiché. A la place de “pdftex,” vous devriez avoir “/usr/lib/pdftex/bin/pdftex”. Il n’est pas nécessaire que “*XTherion*” soit accessible via le “PATH”. Il est possible de le lancer avec son path. Si vous voulez trouver le logiciel dans le “PATH”, vous pouvez ajouter le répertoire, qui contient le programme au “PATH” avec la commande : “export PATH=\$PATH:directory” (pour le shell “bash”, pour les autres shell, la commande est légèrement différente ; “echo \$SHELL” vous indiquera quel shell vous utilisez).

Maintenant allez dans le répertoire de *Therion* et tapez la commande “*XTherion*” (ou “*.XTherion/XTherion*” si vous n’avez pas encore installé *Therion*). Une petite fenêtre s’affiche, pour vous indiquer que *XTherion* est en cours de démarrage. Après un court instant, la fenêtre *XTherion* apparaît comme la figure ci-dessous.



La fenêtre *XTherion* a une barre de menu avec 4 menus : “File”, “Edit”, “Window” et “Help”. Une liste de fonctions est disponible pour chaque menu. Il y a deux zones de travail (toutes les deux vides et noire au démarrage) : celle du dessus pour les fichiers de configuration, celle du fond pour les logs. Le panneau de contrôle est sur la droite, et la barre de statuts au bas de la fenêtre.

### 1.3.1 Help (Aide)

Le menu “Help” a 3 sous menus:

- “Control” ouvre une fenêtre avec le résumé (documentation) des commandes (controls);
- “BAC calculator”: une calculatrice pour el pourcentage d’alcool dans le sang (Blood Alcohol Calculator), le taux métabolique d’élimination de l’alcool (Metabolic Removal Rate (MRR) et le temps que vous devrez attendre avant de dessiner la topographie (ETA, Estimated Time of Arrival);
- “About” affiche la version du programme et les noms des auteurs.

### 1.3.2 File (Fichier)

Comme vous pouvez vous y attendre, ce menu a des sous menus pour créer (“New”) ou ouvrir (“Open”) un fichier de configuration, le sauvegarder (“Save”) et le fermer (“Close”). Il y a aussi un sous-menu pour compiler le projet, “Compile”, et “Exit”, pour quitter *XTherion*.

*XTherion* a plusieurs raccourcis claviers, c.à.d. des touches ou combinaisons de touches équivalentes a des menus ou des commandes. La plus importante est “F9” qui compile le projet. C’est-à-dire qu’appuyer sur “F9” équivaut a choisir le menu “File | Compile”.

### 1.3.3 Window (Fenêtres)

Le menu “Window” alterne les pages de travail de *XTherion*:

- “Text Editor”: Editeur de texte des fichiers de données topographiques (*centerline*);
- “Map Editor”: Editeur du dessin des topographies (scraps);
- “Compiler”: L’éditeur du fichier de configuration et le compilateur du projet. C’est dans ce mode que démarre *XTherion*.
- “Model Viewer”: pour voir en 3D la cavité.

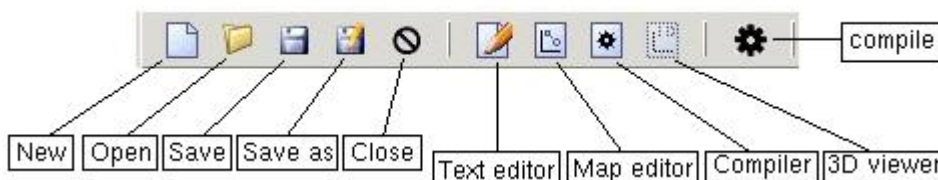
Chaque mode interactif a sa propre barre d’outils (avec son menu spécifique), sa barre de statut (messages de statut spécifiques pour chaque mode), et ses panneaux de contrôle (avec des actions spécifiques pour chaque mode).

Il y a 3 menus pour régler la taille de la fenêtre, pour prendre tout l’écran “Maximize”, ou “Normalize” pour lui donner la taille plus petite modulable à la souris. Enfin pour positionner la page des contrôles sur la fenêtre à gauche ou à droite, “Switch Panel”.

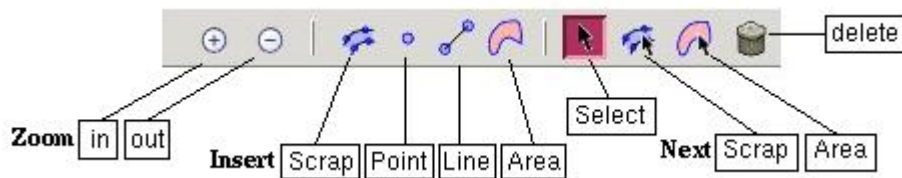
Enfin, il y a le sous-menu “KBD Encoding” pour choisir le type d’encodage du clavier. Si vous voulez en rajouter un, contactez-moi, je le rajouterai ici.

### 1.3.4 Toolbar (barre d’outils)

La barre d’outils de *XTherion* (juste sous le menu) a été introduite avec la version 0.3.10. Elle dispose de boutons pour les différentes opérations comme créer un nouveau fichier, ouvrir un fichier, enregistrer, fermer le fichier, changer de mode de “*XTherion*” et également compiler.



Dans le mode éditeur de carte, la barre d'outils a également des boutons supplémentaires à droite pour les opérations les plus fréquentes : zoom avant et arrière, insertion de scrap, de points, de lignes et de surface, sélection et suppression des éléments.



### 1.3.5 Controls panel (panneau de contrôle)

Le panneau de contrôle est situé par défaut à droite et contient plusieurs boutons sur un fond bleu : ce sont les contrôles. Le nombre et le type des contrôles dépendent du mode interactif. Le panneau peut être déplacé à droite ou à gauche de la fenêtre avec le menu “Window |Switch panels”.

Chaque contrôle peut être ouvert ou fermé : quand il est ouvert, les informations du contrôle sont visibles sous le bouton bleu. Pour ouvrir ou fermer le contrôle, double cliquez sur le bouton.

### 1.3.6 Status bar (barre d'état)

La fenêtre de *XTherion* a une barre d'état au bas de la fenêtre. La barre d'état affiche des informations utiles sur les statuts du programme. C'est pourquoi vous devez prendre l'habitude de lire son contenu.

### 1.3.7 Exercice

Maintenant, essayez de passer d'un mode à l'autre en cliquant sur le bon menu. Notez comment les boutons sous la barre de menu changent. Notez que le libellé du mode apparaît dans le titre de la fenêtre. Sélectionnez le sous-menu “Help” (aide). Vous pouvez également tester le “BAC calculator” ...

Enfin, allez sur la fenêtre du mode “Compiler” .

## 1.4 Essayer *XTherion*

---

Nous allons commencer par travailler sur un exemple, vous pouvez lancer *XTherion* avec un des exemples fournis avec le logiciel. De toute façon, il est préférable de voir en premier lieu comment *Therion* organise les données.

### 1.4.1 Les fichiers de *Therion*

La topographie d'une grotte est un “projet”, avec un fichier de configuration qui indique ce qu'il faut faire, comment le faire, plus des fichiers de données. Il peut y avoir plusieurs fichiers de configuration, afin de calculer les mêmes données de différentes façons et fournir des résultats différents (topo plus ou moins détaillée par exemple).



*Therion* utilise des fichiers texte, que vous pouvez lire et éditer avec n'importe quel éditeur de texte en dehors de *XTherion*. Bien sûr, si vous ne respectez pas la syntaxe de *Therion*, les fichiers ne se compileront pas et *Therion* ne générera pas le document que vous souhaitez.

Pour cette raison, il est préférable d'utiliser *XTherion* pour éditer les fichiers de *Therion*. Il ne fait pas de contrôle de syntaxe, mais il vous aidera beaucoup dans le développement de votre projet topographique.

*thconfig* (Fichier de configuration de *Therion*)

Par convention, le fichier de configuration de *Therion* est nommé *thconfig*. Ce n'est pas obligatoire, et il peut être nommé différemment. Quand vous ouvrez un fichier de configuration dans *XTherion*, par défaut, il affiche uniquement les fichiers qui contiennent la chaîne "*thconfig*" dans leur nom ("*ma-topo.thconfig*", "*thconfig\_grotte*", etc.), mais vous pouvez choisir de sélectionner "tous" les fichiers ("all"). Par convention, les fichiers *Therion* de données topographiques ont l'extension ".th" et les fichiers de cartes ".th2".

### 1.4.2 Exemple

Maintenant créez un répertoire pour cet exemple. Je l'appellerai *gm-therion*. Pour le moment, il est vide, mais il va être rempli de fichiers *Therion* et d'images pour le dessin. Les fichiers générés seront également placés ici plus tard.

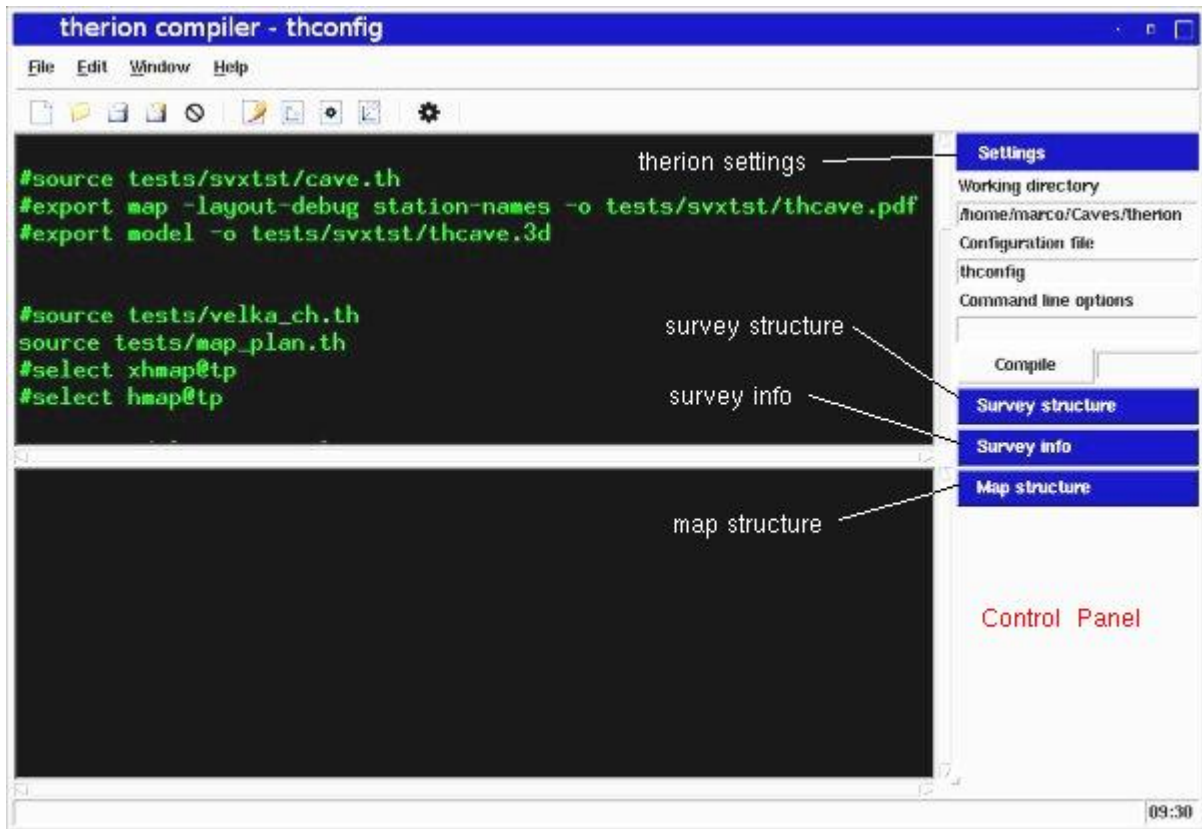
Placez-vous dans ce répertoire et exécutez la commande :

*XTherion*

Après la petite fenêtre de démarrage, qui vous indique que *XTherion* démarre, la fenêtre de *XTherion* apparaît, en mode compilation.

Dans le panneau de contrôle vous trouvez :

- "Settings" (paramètres), avec les paramètres de configuration du programme : "Working directory" (répertoire de travail), le nom du fichier "Configuration file" (fichier de configuration actuel), et d'autres options de la commande *Therion*.
- "Survey structure" (configuration de la topographie) affiche l'organisation hiérarchique de la topographie (découpage en fichiers topo).
- "Survey info" (informations topographiques) affiche quelques informations sur le fichier topographique sélectionné dans la "Survey structure". Les informations sont disponibles après que le projet ait été compilé.
- "Map structure" (organisation des cartes) affiche l'organisation des cartes (calculées).



Vous devez ouvrir un fichier de configuration pour activer ces contrôles. En conséquence, sélectionnez le menu "File | New" et choisissez le nom de fichier propose comme fichier de configuration (nomme "*thconfig*"), puis cliquez sur "Save as". A ce moment, les contrôles sont actifs et vous pouvez voir les paramètres du projet. Mais, si vous essayez de compiler, vous aurez une erreur car vous n'avez spécifié aucune commande de configuration, ni indiqué de données topographiques.

Vous ne pouvez pas non plus exécuter *Therion* avec une option (ou les 2) "-h" (help) ou "-v" (version). Ecrivez l'option dans la zone texte (panneau de contrôle) sous "Command line options" et cliquez sur le bouton "Compile". Le résultat est très court, et le résultat de *Therion* est affiché dans la partie basse de la fenêtre.

### *environnement*

(*Therion*)

Une autre information utile que fournit *Therion* même si le fichier de configuration est vide, est le contexte d'exécution. C'est-à-dire, son environnement. Ecrivez l'option de commande "--print-environment" et compilez. Les variables d'environnement sont

- le fichier d'initialisation : /etc/therion.ini;
- le fichier de configuration du projet;
- INIT: le path (liste de répertoires) ou *Therion* cherche le fichier d'initialisation therion.ini. Ce répertoire est aussi utilisé par *XTherion* pour son fichier d'initialisation, *XTherion.ini*.
- SOURCE: le path des fichiers de données (.th e .th2) et le fichier de configuration ("*thconfig*") pour *Therion*;
- METAPOST : la commande METAPOST;
- PDFTEX: la commande pdfTeX.

Ces informations sont utiles pour résoudre des problèmes d'installation, tel que, par exemple, *Therion* qui ne trouve pas METAPOST.

Maintenant fermez le projet : sélectionnez le menu "File | Close".

### 1.3.3 Une topographie

Il est maintenant temps de voir comment *Therion* travaille. Ouvrez le fichier de configuration des exemples fournis avec le logiciel : sélectionnez le menu "File | Open". Vous avez une fenêtre de sélection de fichier. Allez dans le répertoire d'installation de *Therion* et sélectionnez le fichier *thconfig*. Cliquez sur le bouton "Open". Le fichier apparaît dans la partie supérieure de la fenêtre *Therion*.

Pour le moment, ne vous en préoccupez pas, lancez juste le menu "File | Compile" (ou appuyez sur la touche F9). La zone texte près du bouton "Compile" devient jaune avec le texte "RUNNING". Le résultat de la compilation sera affiché dans la partie basse de la fenêtre. A la fin, si tout s'est bien passé, la zone teste deviendra verte avec le texte "OK". Si quelque chose se passe mal, il deviendra rouge avec le texte "ERROR". Dans ce cas, vous devez identifier le problème et le résoudre. Attention, la zone texte peut afficher "OK" même si des erreurs mineures (warning) se produisent : vous devez parcourir la fenêtre de sortie (partie basse de la fenêtre) avec l'ascenseur pour vérifier qu'il n'y a pas de warnings. Pour les exemples fournis avec la distribution du logiciel, il n'y a pas d'erreurs.

Sur Mac OS X *XTherion* termine toujours la compilation avec le message rouge "ERROR", même s'il n'y a pas d'erreurs. Vous devez vérifier la fenêtre de résultat de compilation pour contrôler qu'il n'y a pas eu d'erreur.

Maintenant vous pouvez ouvrir le fichier "cave.pdf" avec un visualisateur PDF (xpdf ou acroread si vous êtes en mode ligne de commande, sinon double cliquez avec la souris sur le nom du fichier; xpdf a une commande intéressante "reload" qui permet de relire le fichier sans avoir besoin de le fermer), et regardez le résultat.

*Therion* stocke des fichiers temporaires dans le répertoire \$TMP/th\$PID où PID est l'identifiant du processus de la tâche *Therion* (et pas de *XTherion*). Ces fichiers temporaires sont supprimés à la fin de l'exécution.

## 1.5 Configuration de *Therion*

---

*Therion* et *XTherion* se basent sur des variables d'environnement et des fichiers d'initialisation pour configurer leur mode de fonctionnement.

### 1.5.1 Environnement *Therion*

*Therion* utilise les variables suivantes

- THERION, chemin de recherche pour les fichiers d'initialisation;
- HOME, le path \$HOME/.therion est utilisé dans la recherche des fichiers d'initialisation quand ces fichiers ne sont pas trouvés dans le path THERION (ou si ce dernier n'est pas défini);
- TEMP et TMP sont utilisés pour construire le répertoire des fichiers temporaires. Celui-ci est \$TMP/th\$PID, où \$PID est le process id. Ces variables sont aussi utilisées pour le chemin du répertoire de debug où les fichiers de données temporaires sont enregistrées, \$TMP/thDEBUG.

## 1.5.2 Fichiers d'initialisation

Les fichiers d'initialisation, "therion.ini" et "XTherion.ini", sont écrits, respectivement, avec une syntaxe similaire aux fichiers de données. C'est-à-dire avec des lignes de commandes et des commentaires (ces derniers commencent par le caractère '#').

*Therion* cherche les fichiers d'initialisation dans les répertoires dans l'ordre suivants (sous Linux et sous Windows après les deux premiers, les choses sont légèrement différentes) :

1. \$THERION
2. \$HOME/.therion
3. /etc
4. /usr/etc
5. /usr/local/etc

La recherche s'arrête quand le fichier d'initialisation est trouvé. Si le fichier d'initialisation n'est pas trouvé *Therion* utilise des valeurs par défaut (codées dans le programme). Les commandes des fichiers d'initialisation sont :

- "encoding-default", fixe la sortie par défaut d'encodage;
- "encoding-sql", fixe le format d'encodage pour les données exportées dans des scripts SQL;
- "language", fixe la langue pour les textes de légende sur les topographies. Par exemple "language fr";
- "mpost-path", indique le chemin pour METAPOST. Par défaut "mpost";
- "pdftex-path", le chemin pour pdfTeX. Par défaut "pdfetex";
- "source-path", le chemin de recherche pour les fichiers de données (.th et .th2) et le fichier de configuration ("*thconfig*").
- "tmp-path", indique le répertoire où les fichiers temporaires devront être créés. Par défaut "/tmp".
- "tmp-remove", indique la commande pour supprimer les fichiers temporaires.
- "tex-env" utilisée uniquement sous Windows (à faire);
- "tex-font" indique les fontes pour l'encodage. La syntaxe est "tex-font encoding rm it bf ss ssi", où le nom de l'encodage est suivi par la fonte pour les 5 styles : normal, italique, gras, sans-serif, et sans-serif italique.

Le fichier d'initialisation de *XTherion* contient une liste de variables qui modifie le comportement du programme. Les valeurs de ces variables peuvent être fixées aux valeurs par défaut. Parmi celles-ci nous avons :

- le répertoire de travail. C'est le répertoire que *XTherion* affiche par défaut lorsqu'on lui demande d'ouvrir des fichiers. J'ai trouvé pratique de le fixer au sous répertoire "Caves" de mon répertoire de données ("mes documents" par exemple), car c'est ici que j'enregistre toutes mes topo : set xth (gui,initdir) \$env(HOME)/Caves.
- la langue utilisée par l'interface de *XTherion*. Vous pouvez la définir en retirant le commentaire sur la ligne ::msgcat::mclocale XX. Remplace "XX" avec le code de votre langue (fr pour le français). Ce doit être l'une des langues prises en charge par le programme, sinon *XTherion* utilisera l'anglais.

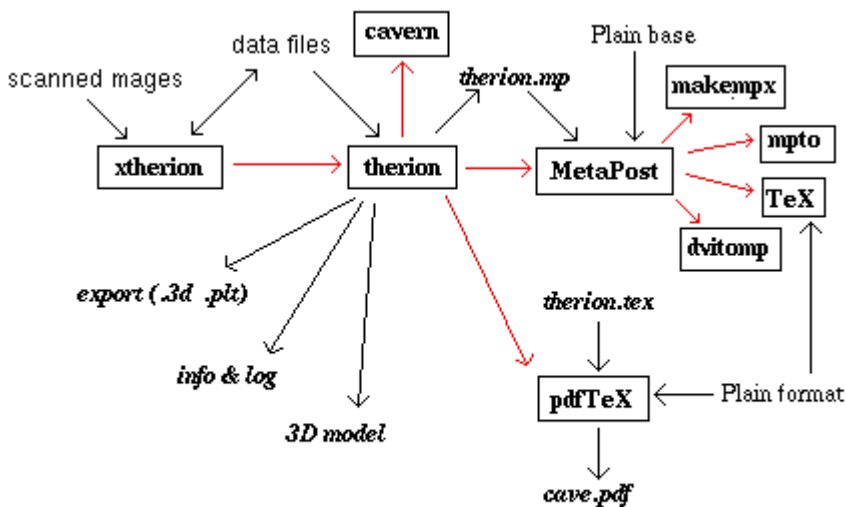
### 1.5.3 Exécution

*Therion* repose sur 3 programmes externes pour compiler les cartes d'un projet:

- cavern, pour calculer les données topographiques;
- METAPOST, pour calculer les scraps (dessins);
- pdfTeX, pour assembler les scraps et construire une topographie.

C'est pourquoi vous pouvez voir dans les résultats de la construction du projet (dans la partie basse de la fenêtre de compilation de *XTherion*) les sorties écran de ces programmes.

*XTherion* sauvegarde les fichiers de données ".th" et ".th2" et le fichier de configuration sur le disque dur lors de la compilation. C'est pourquoi ils sont toujours synchronisés avec leur version utilisée par *XTherion*. Ensuite il appelle *Therion*. En fait, vous pouvez compiler sans passer par *XTherion* et appeler directement *Therion* à partir du shell (la ligne de commande), c'est ce que je fais quand je veux générer une topo d'une série de fichiers sans les éditer.



*Therion* calcul les fichiers de données et coordonne le travail de programmes externes [thbook 8]. Il lit les fichiers de données et interprète les lignes de commande qu'ils contiennent. D'abord il calcul la position des stations topo. En particulier il recherche les bouclages topo et distribue l'erreur sur les différentes visées.

Ensuite il calcul les données des scraps (fichiers .th2) et les transforme suivant la position des stations topo. Après cela, il export les données en fichiers METAPOST (.mp), et appelle le programme mpost. Celui-ci calcul les fichiers ".mp" et produit des fichiers PostScript (.ps). Un petit fichier pour chaque scrap, c'est-à-dire, pour chaque partie de la carte.

Les fichiers PostScript sont calculés par *Therion* et convertis en format pseudo-PDF. Enfin, il appelle pdfetex pour transformer ces fichiers pseudo-PDF. Ce programme produit les topographies en format PDF, en prenant soin du format et des inclusions de texte.

- fr/tbe/wiki1.txt
- Last modified: 11 years ago
- (external edit)

## 2 OBJETS GRAPHIQUES

Dans ce chapitre, nous verrons comment utiliser *Therion* avec des exemples simples. Une petite grotte (ou une portion de grotte). C'est à dire que nous pourrons faire l'ensemble du dessin sur une unique feuille.

Avant que vous commenciez par votre premier exemple, nous devons parler au sujet de la façon dont *Therion* fonctionne, ainsi que la syntaxe du langage. Ceci vous aidera à apprendre les commandes le plus souvent utilisées.

Vous pouvez utiliser des données d'une de vos topographies pour ce tutoriel.

Voici des conseils généraux, pour vous aider commencent à employer *Therion*. Plusieurs de ces points sont expliquées plus tard, mais c'est un bon moment pour mettre en œuvre ces directives.

C'est une bonne idée de découper le dessin d'une cavité branche par branche. Il sera ainsi plus facile de détecter les erreurs et de les corriger. Même si vous êtes en train d'ajouter une galerie à une topographie existante, travaillez sur cette pièce de façon séparée, ainsi vous pourrez corriger plus facilement les erreurs potentielles.

Vous pouvez commencer avec un unique fichier, et entrer les données topographiques, sans dessin. Un unique fichier *thconfig* avec un suffixe ".th" data file(s); sans fichier ".th2". Cela vous donnera une topo avec uniquement le cheminement topographique, et les stations. Avec cela vous pourrez contrôler que vous n'avez commis aucune erreur de saisie des données.

L'étape suivante est d'ajouter un scrap pour le dessin. Concentrez-vous sur une seule carte à la fois. Faites le plan et la coupe développée séparément. Vous pouvez mettre les commandes pour les deux dans le fichier de configuration. Mais commentez celui que vous n'utilisez pas pour l'instant.

Travaillez sur un seul scrap en même temps. Ajoutez les commandes "input" dans le fichier de données topographique, mais commentez celles dont vous n'avez pas besoin. Pour commencer le dessin de la grotte, contentez vous des "station" (points topo) et des lignes "wall" (parois de la cavité). Ne vous souciez pas de connecter les scraps entre eux (avec les commandes "join"). Ne vous précipitez pas non plus sur les éléments de décor qui habilleront votre topographie. Assurez vous que les lignes de contour sont correctement orientées (intérieur/extérieur), que les points topo sont bien positionnes, etc.

À moins que vous soyez un très bon dessinateur de croquis sous terre, il est recommandé que vous n'employez pas les scans de vos croquis levés sous terre comme fonds d'image pour le scrap. Refaites un croquis à partir a partir des minutes topo et de vos croquis réalisées sous terre. Vous scannerez ce nouveau croquis plus précis afin de l'utiliser comme fond d'image pour dessiner le scrap.

### 2.1 Les fichiers *Therion*

---

*thconfig* (fichier de configuration de *Therion*)

Avant que nous puissions aborder un exemple nous devons discuter l'approche de *Therion*. Chaque course de *Therion* dépend d'un dossier de configuration, qui indique quoi et comment faire. Le nom conventionnel pour ce dossier est *thconfig*. Il contient

- la liste des actions réalisées par le logiciel topographique (les fichiers de données);

- les options de présentation des résultats (symboles, légende, etc.);
- les sorties (documents) demandés.

Il y a deux types de fichiers de données :

- données topographiques, nommés conventionnellement avec le suffixe “.th”;
- fichier de dessin, nommés conventionnellement avec le suffixe “.th2”.

Par conséquent les étapes nécessaires pour faire une topographie avec *Therion* sont :

1. préparer les fichiers de données topographiques (éditeur de texte) ;
2. faire les fichiers de données de dessin (éditeur de carte de *XTherion*) ;
3. écrire un fichier de configuration *thconfig* approprié;
4. compiler tout avec le *XTherion*.

Si tout allait bien, le résultat est un fichier pdf que vous pouvez visualiser avec n'importe quel éditeur de PDF. S'il y a des erreurs, celles-ci sont affichées dans la partie inférieure de la fenêtre du compilateur, et vous devez les corriger avant de relancer une compilation.

*TeX* (programme)

*METAPOST* (programme)

*cavern* (programme)

Nous avons vu que *Therion* se fonde sur trois programmes externes : *cavern*, pdfTeX et METAPOST (qui emploie TeX). En effet si vous passiez en revue le résultat de la compilation (partie inférieure de la fenêtre de compilateur), quand vous avez exécuté le fichier d'exemple fournit avec le logiciel, vous noterez les traces de ces programmes.

*XTherion* appelle le programme *Therion*. En fait, vous pourriez réaliser votre topographie sans utiliser *XTherion*, en lançant la commande *Therion* a partir de la ligne de commande. La syntaxe de la commande de *Therion* est décrite dans l'annexe. 2.c.

### 2.1.1 Fichier de configuration

Le fichier de configuration *thconfig* contient le nom du fichier de donnée (avec le chemin relatif a son répertoire), utilisez la commande

```
<code>
  source file.th
</code>
```

Il peut contenir des données définissant le positionnement. Par exemple,

```
<code>
  layout xvi_export
  scale 1 500
  grid bottom
  grid-size 10 10 10 m
```

```
endlayout
</code>
```

### *grid* (commande d'affichage)

La commande `grid` indique pour dessiner la grille sur la carte. La valeur le “bottom” indique pour la mettre sur la plus basse couche de sorte qu'elle ne couvre pas le dessin de la grotte. Si au lieu du “bottom” vous écrivez le “top” la grille apparaît au-dessus du schéma [thexample 6]. La commande `grid-size` définit la dimension de grille, dans X, Y et Z. Vous pouvez également donner les coordonnées d'origine de grille avec la commande `grid-origin` de suivi des trois coordonnées [thbook 40].

Le fichier de configuration contient les commandes qui indiquent quels résultats doit-on créer. Par exemple,

```
<code>
export model -fmt survex
export map -fmt xvi -layout xvi_export -o file.xvi
export map -proj extended -fmt xvi -o extended.xvi
export map -proj plan -o plan.pdf
export atlas -proj plan -o plan_atlas.pdf
</code>
```

Pour plus de détails sur la syntaxe du fichier de configuration, se reporter à l'annexe 2.a.

## 2.1.2 Données topographiques

### *SVX* (Format des données *Survex*)

Les fichiers de données topographiques (extension .th) ont une syntaxe très proches du logiciel `survex` [1](#), mais il y a quelques modifications importantes :

- Les commentaires commencent avec '#' au lieu de ';'.
- Les mots clés ne débutent pas par '\*'.
- A la place de “begin”-“end” *Therion* utilise `survey-` `endsurvey`.
- *Therion* référence les stations (points topos) par “`station@survey.cave`” au lieu de “`cave.survey.station`”.
- Il n'y a pas de commande “`export`”.
- Les données topo doivent être regroupées dans un bloc `centerline-` `endcenterline`.
- Le séparateur décimal est le point '.'.
- La commande “\*`title`” est remplacée par l'option “-`title`” de la commande `survey`.

La syntaxe des fichiers de données *Therion* est décrite dans l'annexe. 2.b qui liste également les commandes de dessin. Pour commencer, vous pouvez débiter avec les commandes suivantes:

- Un fichier topographique commence avec la commande “`survey`”. Elle est suivit par l'ID topographique, et de façon optionnelle, par le titre, sur la même ligne. Par exemple: “`survey gm0 -title Grazie_Mille`”. La commande “`survey`” est comme une parenthèse ouvrante. Elle doit être fermée par la commande “`endsurvey`”.
- Dans le bloc “`survey`”-“`endsurvey`” il peut y avoir un ou plusieurs `centerline-` `endcenterline`), et d'autres blocs `survey` (topos). Dans le bloc “`centerline`”-“`endcenterline`” vous pouvez intégrer des données topo et d'autres commandes.



- La commande “data” détermine le type et l’ordre des données dans les lignes de données topographiques. Exemple: “data normal from to length compass clino”.
- La commande “fix” définit les coordonnées géographiques d’une station topo. Exemple: “fix 1 575628 4476124 1250”. Vous pouvez répéter le “fix” pour plusieurs stations, fixant ainsi plus de points.
- La commande “date” indique la date de la topographie. Exemple: “date 2002.05.18” indique le 18 Mai 2002.
- La commande “team” indique le nom des membres de l’équipe topographique et leurs rôles. Exemple: “team “S./Mudrak” compass clino”.
- La commande “units” indique les unités de mesure pour chacune des valeurs. Par exemple “units length 0.01 meters” signifie que les longueurs sont en centimètres. Par défaut les distances sont en “meters” (mètres), et les angles en “degrees” (degrés).
- La commande “declination” indique la correction de la déclinaison magnétique. Dans un “survey” elle s’applique à toutes les mesures (tous les blocs “centerlines”); sinon, mettez la dans un bloc “centerline” (pour la limiter a ce bloc).
- Les données topo sont écrites sur une même ligne pour une même visée topo et dans l’ordre des valeurs définies dans la commande “data”. Comme symbole décimal utilisez le point ‘.’
- Les commentaires débutent avec le caractère ‘#’

Le fichier de données survey doit avoir un bloc *survey/endsurvey* intégrant toutes les commandes, excepté la commande encoding. Par conséquent le survey est un conteneur de tous autres objets. Comme il peut contenir également l’autre survey, il est possible de structurer les données topo (y compris les dessins) hiérarchiquement. Le survey est comme un répertoire dans un disque dur. Comme un dossier il a un nom, qui est employé pour se rapporter au survey. Le nom du survey (l’ID) est écrit après que la commande de “survey” sur la même ligne [thbook 12].

### *centerline (commande de Therion)*

Un bloc survey peut contenir, un bloc centerline/ endcenterline , la commande input inclue un autre fichier de données (un fichier de dessin, ou un fichier de topographie).

```
<code>
encoding UTF-8
survey nome -title "cave title"
  centerline -extend left
    title "survey title"
    date 2006.10.26
    team "nome/surname" notes
    team "nome/surname" clino compass tape
    data normal from to length clino compass
    1 2 4.37 12.5 321
    2 3 12.56 10 276
    ...
    ...
  input file.th2
endcenterline
endsurvey
</code>
```

La commande *centerline* peut aussi être écrite *centreline*.

### *extend* (commande de *Therion*)

La commande *centerline* dans l'exemple ci-dessus est complétée par l'option “-extend left”. Ceci indique pour dessiner coupe développée de droite à gauche. Par défaut *Therion* la dessine de gauche à droite. Par conséquent, vous pouvez écrire la commande *extend* suivie de “left” or “right” au milieu des lignes de données topographiques et les dessins des visées seront reproduits vers la gauche ou la droite suivant les commandes. Si la commande est suivie de « normal” ou “reverse” les visées suivantes seront tracées dans le même sens que la précédente, ou dans le sens opposé [thbook 16]. Il y a également “extend ignore” pour les visées qui ne devraient pas apparaître dans la coupe développée.

La commande *extend* doit être définie suivant l'orientation de la cavité désiré sur la coupe. Si la galerie dans le croquis va vers la droite et vous écrivez “extend left” dans vos données le résultat ne sera probablement pas celui que ce que vous escomptez.

### *input* (commande de *Therion*)

La commande *input* est utilisée pour inclure un autre fichier de données. Dans l'exemple ci-dessus, elle incluse un fichier de dessin (extension .th2). Si le fichier avait l'extension .th, celle-ci aurait pu être omise [thbook 12].

## 2.1.3 Drawing data

Les fichiers avec les données de dessin (appelées *scrap*) sont des fichiers textes avec des commandes pour des points, les lignes, les surfaces, les symboles, etc. Ces commandes sont interprétées par *Therion* quand vous compilez le projet pour produire les données destinées à METAPOST.

Les commandes de dessin sont énumérées dans l'annexe 2.b, et elles seront décrites au fur et à mesure dans la documentation *Therion* en utilisant des exemples.

## 2.2 *XTherion*

---

Démarrons avec un exemple très simple : un fichier de configuration et un unique fichier de données topographiques. Comme les fichiers de *Therion* sont des fichiers texte, vous pouvez les éditer avec n'importe que éditeur de texte, mais cela sera plus facile d'utiliser *XTherion*.

L'avantage de *XTherion* est que pendant que vous l'utilisez, vous pouvez essayer et compiler votre projet, et s'il y a des erreurs, vous pouvez les corriger. *XTherion* vous aide en surlignant les messages d'erreur dans la fenêtre de sortie standard (fenêtre en arrière plan) et en les associant à la ligne en erreur dans le fichier source.

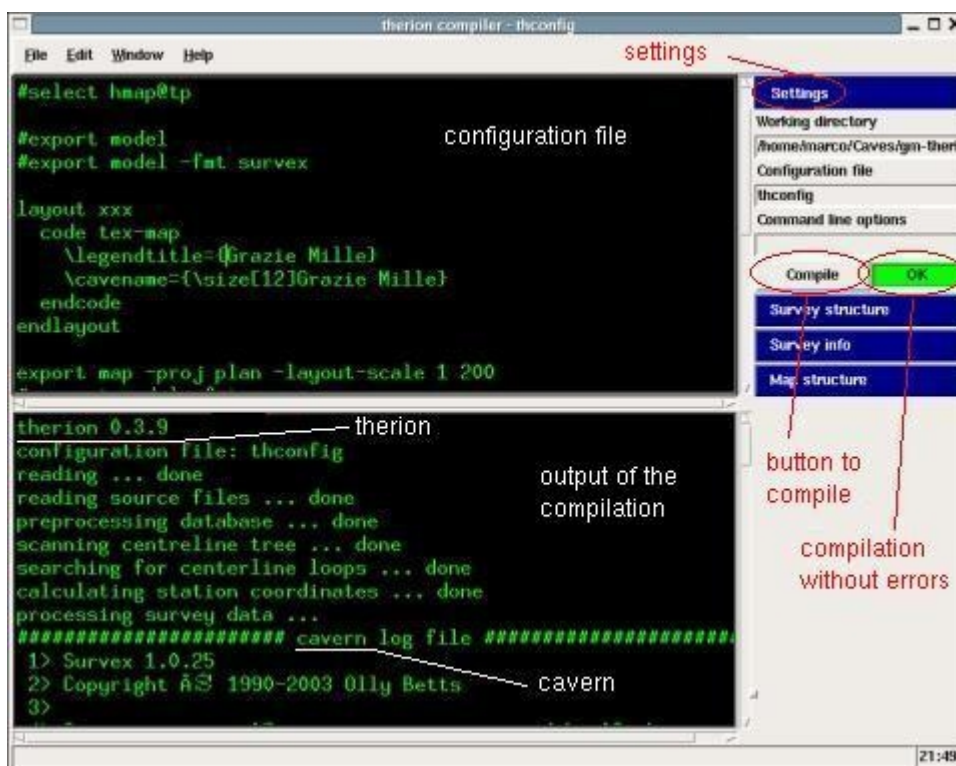
Maintenant exécutez *XTherion* qui démarre avec le mode compilateur. Vous pouvez créer un nouveau fichier de configuration (menu “Fichier | Nouveau”), ou ouvrir un fichier existant. Créez un répertoire pour cet exemple et déplacez-vous dans ce dossier. Mettez ces fichiers exemple dans ce dossier.

- *thconfig*, fichier de configuration
- *gm0.th*, fichier de données topographiques
- *gm0.th2*, fichier de dessin

Ouvrez le fichier “*thconfig*” avec *XTherion* (menu “Fichier | Ouvrir”). Il indique qu’il faut utiliser le fichier de donnée “*gm0.th*”, il indique un fichier de dessin (nommé “xxx”) avec deux lignes de codes pour TeX, et deux commandes d’affichage (“*export*”); la première pour la coupe développée (scale 1:200), la seconde pour le plan, avec plusieurs options de présentation.

Maintenant, passez de *XTherion* en mode “Texte Editeur” et ouvrez le fichier de donnée (menu “Fichier | Ouvrir”). C’est presque un fichier de données topographiques, avec une syntaxe légèrement différente, mais vous ne devriez pas être trop dérouté pour la comprendre. Si vous avez des difficultés, reportez vous à la syntaxe des données *Therion* dans l’annexe 2.b.

Maintenant vous pouvez tout compiler (menu “Fichier | compiler”. “*XTherion*” bascule en mode “compilateur” et, à côté du bouton, apparaît sur le panneau, en jaune “*RUNNING*”. Après un moment il devrait se transformer en un affichage vert “*CORRECT*”. S’il s’affiche le message rouge “*ERREUR*” cela signifie que quelque chose a mal tourné : examinez la fenêtre de sortie pour vérifier les messages d’erreur. S’il n’y a aucun message d’erreur (et que *Therion* a produit un fichier pdf) que l’erreur pourrait être due à un programme absent (ou une installation incorrecte, ou une erreur d’exécution). Assurez-vous que *Therion* peut trouver les programmes dont il a besoin (*cavern*, *metapost*, et *pdftex*).



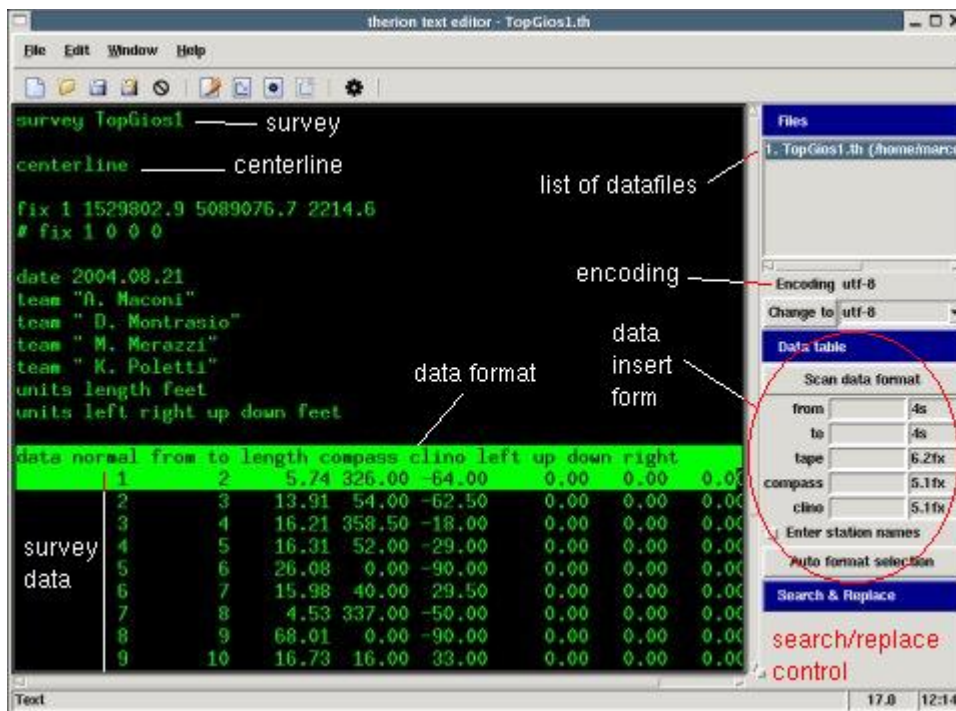
Vous pouvez visualiser le fichier pdf avec un programme pdf viewer. Il ne contient pas grand-chose : seulement les segments aux positions des stations topographiques et quelques lignes d’entête. Après tout, les données fournies se résument aux données topographiques et nous n’avons pas encore réalisé de dessin.

Avant de continuer, faites encore quelques essais, ajoutant les différentes options d’affichage (layout), pour voir comment le pdf change. Reportez vous à l’Annexe 2.a pour plus de détails dans les commandes du fichier.

## 2.3 Éditeur de Texte

L'éditeur de texte de *XTherion* autorise d'écrire et d'éditer (et de sauver) les données topographiques. Ces fichiers doivent être écrit en utilisant la syntaxe *Therion* pour être utilisé par *Therion*, voir l'annexe 2.b.

Pour passer sur l'éditeur, sélectionnez le menu "Window | Text editor". Puis, ouvrez un fichier de données topographiques (extension .th) avec le menu "File | Open", ou créez un nouveau fichier avec le menu "File | New".



### 2.3.1 Panneau de Contrôle

Le panneau de contrôle de l'éditeur de texte contient

- La liste des fichiers ouverts dans le *Files* control. Il est possible d'ouvrir d'autres fichiers que ceux affichés. Cliquez sur le nom d'un fichier pour l'ouvrir automatiquement dans la fenêtre d'édition.
- Le fichier en cours de modification avec les options de modification possibles;
- Un fenêtre de saisie des données *Data table*;
- Un panneau de *Search and Replace* pour effectuer des recherches dans la fichier de données en cours d'édition.

### 2.3.2 La "Data table"

La "Data table" du panel de contrôle simplifie la saisie des données topographiques [thbook 29]. Après avoir saisi la longueur, vous passez à la donnée suivante ("compass") en tapant "Enter" ou la touche "Tab" du clavier. De la même façon, vous passez au "clino". Enfin, pour insérer les données dans le fichier, vous pressez "Enter". Si vous appuyez sur "Tab" le curseur se déplace sur la case "from".

Pour chaque donnée insérée, la fenêtre "Scan data format" est mise à jour pour la nouvelle entrée. Par exemple, si vous avez juste tapé la donnée du point topo 16-17, la zone "from" devient "17" et "to" devient

“18”. Si la case “Enter station names” est cochée, le curseur est positionné sur l’entrée “from” (la valeur est également mise à jour). Si la case n’est pas cochée, le curseur se positionne dans l’entrée “length”.

L’entrée sur la colonne de gauche définit la précision des données, c’est-à-dire que le nombre indique le nombre de décimales à écrire dans le fichier. Par exemple “6.2fx” indique que la longueur doit avoir 2 chiffres décimaux (en plus de 6 positions entières). Pour le nom des stations topo, “.6s” indique d’écrire dans le fichier texte 6 caractères du nom de la station.

### *scan data*

Le bouton “Auto format selection” devrait positionner automatiquement les formats en fonction des données saisies. Le fonctionnement est le suivant

1. écrire (ou copier/coller) la ligne de format de données;
2. Appuyez sur la touche “[Enter]”;
3. Quand le curseur est juste sous la ligne de format de données pressez le bouton “scan data format”
4. saisissez les données topographiques.

D’après to S. Mudrak,

D’abord, appuyez sur le bouton “scan data format”. Cela met à jour le format > au style de vos données. Vous pouvez alors modifier le format des chaînes (du format affiché > en suivant la syntaxe de la fonction C printf() ), vers le format des données dans votre table.

Faites attention, les données saisies dans la table sont insérées dans le fichier (ouvert en édition) à la position où se trouve le curseur. Donc, avant d’utiliser la “Data Table” mettez le curseur dans une ligne blanche, dans le bloc “centerline”. L’insertion automatique ajoutera une nouvelle ligne après chaque ligne de donnée.

### *search*

#### **2.3.3 Recherche et Remplace**

La fenêtre d’éditeur de texte, comme pour tout éditeur de texte, a un panneau pour faire de la recherche/remplacement de texte. Optionnellement elle peut rechercher dans tout le fichier. Les fichiers *Therion* sont des fichiers texte ; même les fichiers de dessin sont écrits en commandes texte : c’est pourquoi les recherches dans les fichiers sont toujours des recherches de texte.

Dans l’éditeur de texte, le résultat d’une recherche est affiché dans le fichier. Le mot (ou les mots) retrouvé(s) est surligné en rouge dans le texte. Pour l’éditeur de cartes, le résultat est affiché dans la fenêtre de dessin, l’élément graphique trouvé est surligné.

Ces fonctions sont décrites dans la section 4.4.

## **2.4 Éditeur de carte**

---

Dans le mode éditeur de carte *XTherion* peut écrire, éditer et sauver des fichiers avec des données de dessin, appelés fichier dessin (map files). La syntaxe de ces fichiers n’est pas compliquée. Mais il est conseillé d’utiliser l’éditeur de dessin plutôt que d’écrire les commandes avec un éditeur de texte à cause des valeurs numériques des commandes. Vous devez vous rappeler que *XTherion* n’est pas un logiciel

WYSIWYG (ce que vous voyez est ce que vous obtiendrez) et il affiche des points et des lignes là où au final, vous obtiendrez des symboles, labels et ainsi de suite. L'éditeur de carte gère uniquement la partie graphique des commandes de *Therion*. Mais de toute façon, le fichier dessin peut être édité avec l'éditeur de texte de *XTherion*, ou n'importe quel autre éditeur de texte.

Pour basculer dans le mode éditeur de carte, allez dans le menu "Window | Map editor". La fenêtre de *XTherion* affiche une zone de dessin et une zone de contrôle (sur le côté). Celui-ci contient les objets suivants:

- "Objects": liste des éléments du fichier de dessin
- "Search and select": pour rechercher un élément graphique dans le fichier de dessin
- "Command preview": Ligne de commande telle qu'elle apparaît dans le fichier de dessin
- "Points": point présent sur le dessin
- "Lines": ligne présente sur le dessin
- "Line points": Liste des points d'une ligne
- "Areas": surface présente sur le dessin
- "Scraps": portion de dessin (présent dans le fichier dessin)
- "Text editor": éditeur de texte pour insérer du texte à la main dans le fichier de dessin
- "Drawing area": zone de dessin
- "Background images": zone de saisie pour l'image d'arrière-plan de la fenêtre de dessin (cette image ne sera pas incluse dans la sortie finale).

L'éditeur de dessin a deux modes d'opération : insertion et section. Vous pouvez passer du mode "insert" à "select" en appuyant sur la touche échappe ("Esc"). Vous passez de "select" à "insert" quand vous sélectionnez un des menus "Edit | Insert" (ou avec un raccourci clavier équivalent). Suivant les menus, vous insérez des points, des lignes ou des surfaces. L'insertion d'un scrap ou de texte laisse l'éditeur de carte en mode "select".

### 2.4.1 Faire un dessin

En sélectionnant le menu "File | New" la zone de dessin devient jaune et elle est prête pour dessiner. Commencez par définir un nom pour le fichier : menu "File | Save as".

Pour ouvrir un fichier th2 existant, utilisez le menu "File | Open", ou "File | Open (no pics)" si vous voulez ouvrir sans charger l'image de fond (ce qui est plus rapide).

Le menu "Edit" a des sous-menus spécifiques pour l'éditeur de dessin,

- "Insert", qui offre le choix parmi "point", "line", "area", "text" and "scrap"
- "Insert image" pour charger une image de fond.

*zoom (Edit menu)*

Vous pouvez déplacer le dessin avec un clique-et-drag du bouton droit de la souris. Vous pouvez changer le zoom, en sélectionnant un nouveau zoom dans le panneau "Drawing area", ou avec le menu "Edit | Zoom ...".

*background*

### 2.4.2 Image d'arrière-plan

*xvi (Therion vector format)*

*tkimg (Tcl/Tk extension)*

Maintenant ouvrez une image d'arrière-plan (menu "Edit | Insert image"). *XTherion* peut lire des images en format GIF, PNM, PPM, et XVI (xvi est le format vectoriel de *Therion*). Si vous avez l'extension tkImag de Tcl/Tk installée, *XTherion* peut également lire les images PNG et JPEG.

REMARQUE : le programme doit être capable de lire des images présentes dans n'importe quel répertoire. Cependant, j'ai découvert qu'elles devaient être présentes dans le répertoire du projet, celui qui a le fichier *thconfig*.

Vous pouvez également charger une image d'arrière-plan avec le bouton "Insert" du panneau "Background images". L'image d'arrière-plan ne fait pas partie du dessin, mais elles sont chargées dans *XTherion* et sont utilisées pour dessiner les scraps. Pour les retirer, utilisez le bouton "Remove" du panneau "Background images". Vous pouvez également les retirer temporairement en utilisant la case à cocher "visibility", afin d'économiser de la mémoire sur l'ordinateur, si vous en avez besoin.

Vous pouvez avoir plus d'une d'images d'arrière-plan en mémoire en même temps, sélectionnez en une et mettez-la au devant des autres ("*Move front*") ou en arrière ("*Move back*"). En utilisant le curseur vous pouvez la rendre plus sombre (déplacez le curseur vers la gauche) ou plus claire (vers la droite). Finalement, vous pouvez déplacer indépendamment chacune d'elles sur le fond de dessin, en utilisant le bouton "*Move to*".

Il est possible de faire tourner et redimensionner chacune des images individuellement. Pour cela il est conseillé de préparer les images scannées avec la même orientation et la même résolution (pixel par mètre, qui représente le même nombre de DPI si les images ont la même échelle) [thbook 29].

#### *auto adjust* (Edit menu)

Le menu "Edit | Auto adjust" (ou le bouton analogue du panneau de dessin "Drawing area") calcul la dimension de la zone de dessin, en utilisant les images d'arrière-plan, et en les redimensionnant dans le canevas. [thwiki 2]. Après chargement d'une image d'arrière plan, sélectionnez ce menu pour l'afficher dans la zone de dessin.

### 2.4.3 Un scrap

Un plan de grotte est composé de scraps, et chaque élément de la carte doit être contenu dans un scrap [thbook 17]. Un scrap est une partie du dessin correspondant à une portion de la grotte sans galeries superposés. Pour faciliter la connexion entre les scraps pour former l'intégralité du plan de la grotte, il est conseillé de découper la carte en scraps (portions de claques) là où la cavité est "simple" (parois pleines, sans particularités devant être dessinées telles que puits, flaques d'eau, concrétions, ...).

Maintenant insérez un scrap: sélectionnez le menu "Edit | Insert - scrap", ou le bouton "Insert scrap" du panneau "Objects", ou le raccourci clavier Ctrl-R. Vous devez observer que le panneau "Command preview" a une nouvelle ligne qui débute par "scrap scrap1 ...". Ce panneau est pratique pour voir les commandes que *XTherion* écrit dans le fichier de données, pour chaque élément de dessin [thwiki 10]. Si vous contrôlez le panneau "Objects" vous voyez 3 lignes

```
scrap
endscrap
end of file
```

Tous les éléments qui appartiennent à ce scrap seront placés entre ces deux premières lignes.

Vous pouvez mettre plus d'un scrap dans un même fichier de données. Dans ce cas, vous aurez plus d'une section "*scrap / endscrap*" dans le panneau "Objects". Il est important d'insérer les éléments de

dessin quand un scrap est sélectionné. Cet élément ira alors dans le bloc du scrap. Quand vous ajoutez un scrap, il est automatiquement sélectionné. C'est pourquoi vous pouvez insérer les éléments de dessins directement après avoir inséré le scrap. Ceux-ci iront directement dans ce dernier.

Le panneau de contrôle du "scrap" vous permet de changer ses propriétés. Les plus importantes sont le type de projection et son id (le nom du scrap).

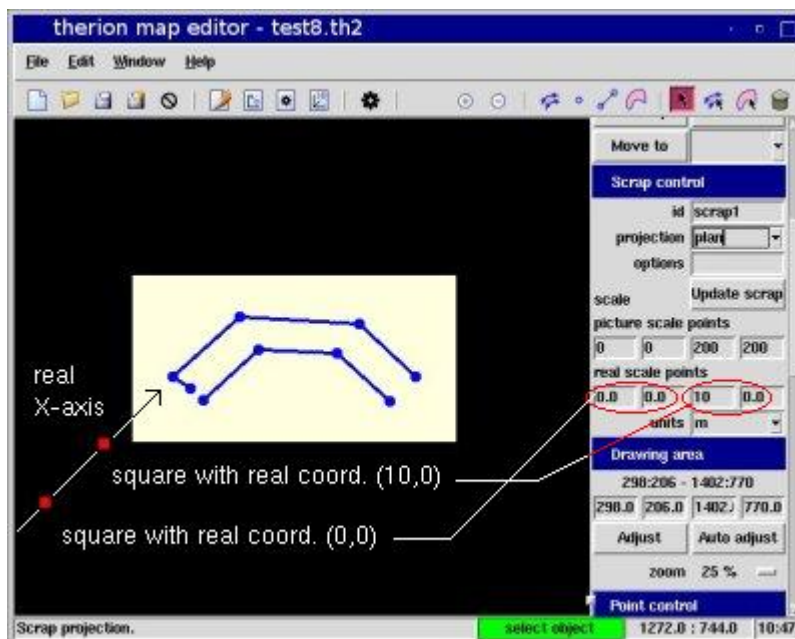
Dans le canevas, il y a deux petits carrés rouges, dans la partie basse, sur la droite et sur la gauche. Ils servent de référence pour le scrap quand il n'y a qu'une station topo dans le scrap, comme par exemple dans le cas d'une section de galerie. Pour définir l'échelle du scrap, vous devez déplacer les carrés aux coins opposés d'un rectangle (idéal) qui contiendrait tout le dessin. Par exemple, celui de gauche dans le coin en bas à gauche, et le droit dans le coin en haut à droite. Puis vous devez fixer les coordonnées de ces deux points (exprimés en mètres si vous utilisez le mètre comme unité de mesure) : par exemple (0,0) pour celui de gauche, et (largeur, hauteur) pour celui de droite.

La position de ce rectangle est transposée dans le fichier ".th2" dans les options de la ligne de commande du "scrap"

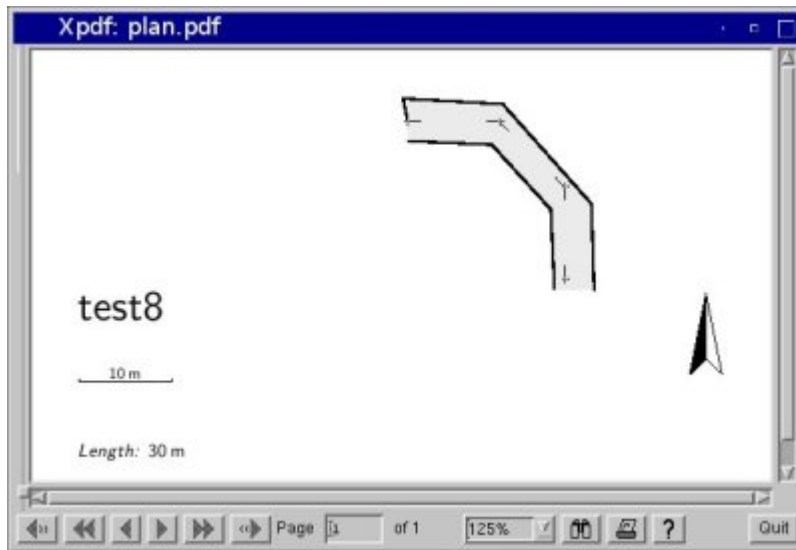
```
-scale [0 0 200 200 0.0 0.0 10 0.0 m]
```

Les quatre premiers nombres sont les coordonnées X1,Y1,X2,Y2 sur le dessin. Les quatre nombres suivants sont les coordonnées sur la carte, suivies par l'unité de mesure.

La position du rectangle définit également l'orientation du dessin. Par exemple, si le rectangle est placé diagonalement dans le dessin et que l'on assigne les coordonnées (0,0) et (10,0), le dessin sera pivoté de 45 degrés dans le sens des aiguilles d'une montre, comme le montre la figure ci-dessous.







Si la coupe a une échelle différente du plan (ou de la coupe développé), vous devez utiliser des coordonnées remises à l'échelle. Par exemple, si le plan est au 1 :1000 et les sections sont au 1 :500 vous devez doubler les valeurs des coordonnées (du rectangle).

## 2.5 Les Points

---

Un point est une position dans le plan où *Therion* dessine un symbole spéléologique, une étiquette, une station topo, etc.

Les stations topo sont des points spéciaux. Elles sont très importantes car elles permettent de mettre le dessin à l'échelle en fonction des dimensions des visées présentes dans les centerline des fichiers de données.

### 2.5.1 Les stations

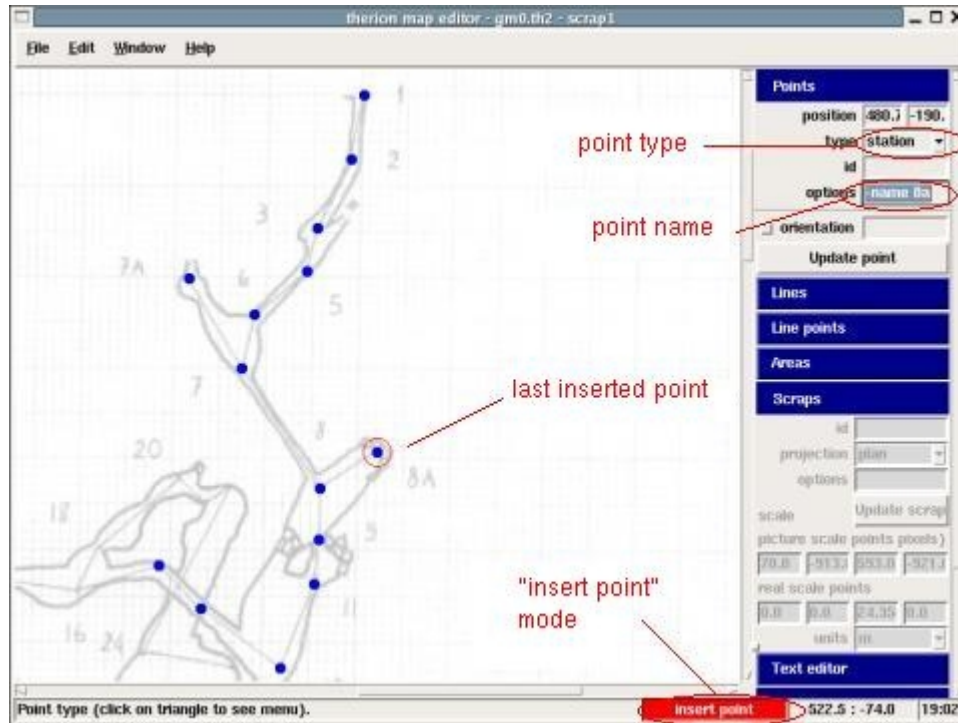
#### *station (type)*

Pour associer un dessin à une *centerline*, il est nécessaire d'insérer des points stations (c.à.d. des points de type "station") dans le scrap. Sélectionnez le menu "Edit | Insert - point" et cliquez avec le bouton gauche de la souris sur la position de l'image de fond où la station est positionnée. Il est important que l'image de fond soit parfaitement dessinée. *Therion* fera attention à étirer et adapter la station en fonction des données du centerline (c.à.d. la position réelle mesurée par les instruments).

Quand vous êtes en mode "insert point", la barre de statut affiche en rouge un message "insert point". Pour vérifier que vous n'avez pas fait d'erreur, vous pouvez compiler (menu "File | Compile", ou touche F9) et *XTherion* affiche la fenêtre de compilation et compile le projet. Si tout va bien, vous pouvez voir le message en vert "OK", et *XTherion* revient dans la fenêtre d'éditeur de dessin.

Dans le panneau "Points" vous pouvez choisir le type de point(s) que vous voulez insérer. Une fois sélectionné, le type reste inchangé pour tous les points que vous insérez, jusqu'à ce que vous le changiez à nouveau. C'est pourquoi il est pratique d'insérer en groupe tous les points d'un même type.

Maintenant, insérez quelques points de type station et positionnez leur option "-name N" où N est le nom de la station dans le fichier de données topographique. Comme nous l'avons dit *XTherion* préserve le type du dernier point inséré, c'est pourquoi il est pratique d'insérer toutes les stations, les unes après les autres. D'autant plus que *XTherion* met à jour automatiquement l'option "-name N" à chaque nouvelle insertion en incrémentant la valeur de N. C'est pourquoi vous devez faire attention et la corriger quand le numéro de la station change brutalement.



## 2.5.2 Autre types de points

Il y a plusieurs autres types de points. Parmi ceux-ci :

- Symboles spéléo : [air](#) (courant d'air), [water](#) (eau), [source](#) (source), [sink](#) (perte),
- Symboles géologiques: [anastomosis](#) (anastomose), [breakdown](#) (trémie, effondrement), [block](#) (blocs), [rock](#) (rocher), [clay](#) (argile), [calcite](#) (calcite), [snow](#) (neige), ice (glace), [scallop](#) (coups de gouge), [sand](#) (sable), [pillar](#) (pillier), [karren](#) (lapiaz), [pebbles](#) (galets), [gypsym](#) (gypse),
- dépôts: [aragonite](#) (aragonite), [pearl](#) (perles), [crystal](#) (cristaux), [disc](#) (disque), [stalagmite](#) (stalagmite), [stalactite](#) (stalactite), [soda straw](#) (fistuleuses), [helictite](#) (excentrique/hélicite), [popcorn](#) (choux-fleur), [column](#) (colonne), moonmilk (mondmilk)
- ...: [height](#) (hauteur), [archeo-material](#) (matériel archéologique), [paleo-material](#) (matériel paléontologique), [camp](#) (camps), [date](#) (date), [section](#) (section), [root](#) (racine), [note](#) (note), [label](#) (libellé, étiquette)
- [centerline](#): station (point topo)
- ...: [anchor](#) (ancrage), [traverse](#) (traversée), [rope](#) (corde), [ladder](#) (échelle), [entrance](#) (entrée), [narrow-end](#) (passage impénétrable), [low-end](#) (passage bas), [continuation](#) (suite possible)

Les points sont visualisés sur le fond du dessin par des points bleus. Le dernier point inséré est toujours le point sélectionné : il est affiché avec un cercle rouge autour du point bleu. Pour changer de point

sélectionné, sortez du mode "insertion" (touche "Esc" ou menu "Edit | Select"), et cliquez sur le point que vous souhaitez sélectionner.

Pour déplacer un point après l'avoir sélectionné avec le bouton gauche de la souris, il suffit de le faire glisser (drag) avant de relâcher le bouton. Pour supprimer un point, sélectionnez le point et pressez Ctrl-D (ou le menu "Edit | Delete").

### *Orientation*

Si le point a une orientation (par exemple, les points de type "gradient" (flèche pour indiquer la pente) "water-flow" (arrivée d'eau), "air-draught" (courant d'air), ...) il est affiché avec une flèche jaune. Par défaut l'orientation est de 0 (vers le nord, ou vers le haut). Pour changer l'orientation de ce point, tracez la flèche (ou écrivez la valeur en degrés dans la case appropriée du panneau "Points").

### *scale (option)*

Il est possible de changer la taille du symbole avec l'option "-scale". La taille par défaut est "m" (normal). Les autres valeurs sont "s" (petit), "xs" (très petit), "l" (grand), "xl" (très grand).

### *clip (option)*

Pour la plus part des points, il est possible de spécifier (option "-clip") de couper ou non le symbole du point en dehors du tracé de la cavité ou non.

### *visibility (option)*

Chaque point peut être masqué avec l'option "-visibility off" (par défaut elle est à "on").

### *text*

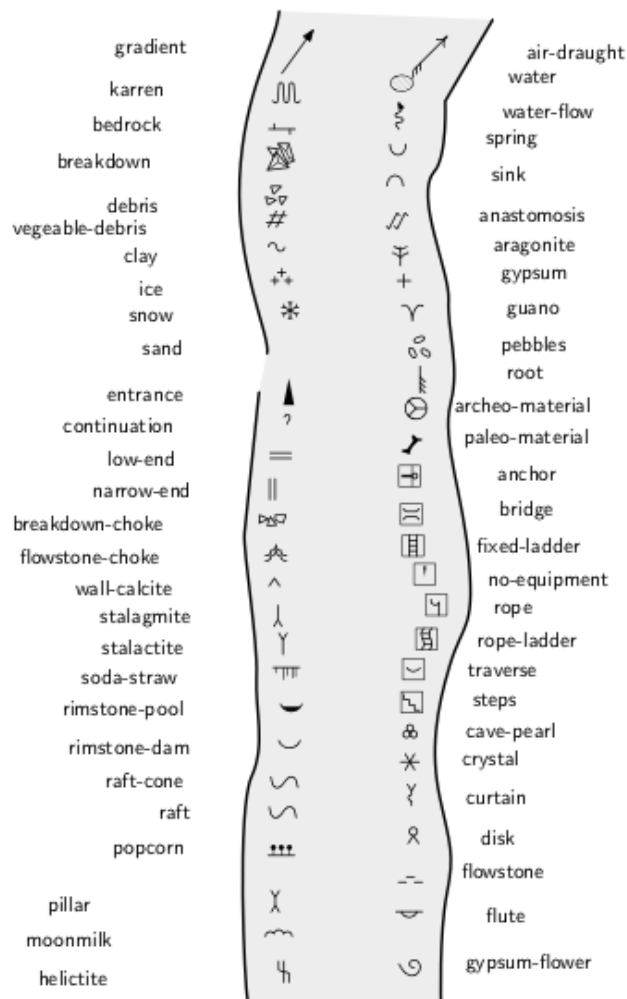
Certains types de points doivent avoir une option "-text". Les points de type "label" doivent avoir un texte (option "-text") et peuvent être orientés (sur une figure le libellé de l'étiquette est orienté à 45 degrés). Les points de type "survey-name" ont une option "-text" avec comme texte le nom de la station.

### *value (option)*

Plusieurs types de points ont une option "-value". Les points de type "date" doivent avoir une valeur (option "-value") sinon, rien ne sera affiché sur la carte finale. Les points de type "dimension" ne sont pas matérialisés graphiquement, mais ils contiennent l'information de la dimension de la cavité (hauteur de galerie), haut et bas si le scrap est un plan ; par exemple, l'option "-value 4 2" signifie une dimension de 4 (m) vers le haut et 2 vers le bas.

Les points de type "height" doivent avoir une "-value" qui peut être positive (chimney : cheminée) ou négative (pit : puits), et peuvent avoir un point d'interrogation facultatif; par exemple "-value [20? m]". Les points de type "passage-height" (hauteur de la galerie ou de la zone) doivent avoir une "-value" qui signifie la hauteur de la galerie (valeur positive, ex, "+4.5"), ou la profondeur du sol, ou du siphon (valeur négative), ou la distance floor-ceiling (sol-plafond)(un nombre sans signe + ou -), ou une paire de nombres (distances vers le plafond et vers le sol, ex, "[5.6 1.2]"). Un point de type "altitude" a une "-value" qui indique l'altitude qui doit être en corrélation avec l'altitude de la station topo la plus proche. Pour indiquer une altitude absolue, utilisez le mot clé "fix", ex, "-value [fix 2126]". Un point de type "dimension" a une "-value" qui indique les dimensions au dessus et en dessous du centerline, ex, "-value [2.6 1.5]".

Le dessin ci-dessous, montre les différents points symboles dessinés avec les symboles UIS (qui sont ceux par défaut). Les points dans d'autres jeux de données ne sont pas très différents.



### Subtype

Certains points ont aussi des sous-types. Le type "water-flow" peut avoir le sous-type "permanent" (défaut), "intermittent", et "paleo". Les points de type "station" peuvent avoir un sous-type "temporary" (temporaire) (par défaut), "painted" (peint), "natural", et "fixed".

## 2.6 Les lignes

### Lines

Le dessin d'une topographie de grotte n'est pas fait uniquement de symboles (points) et de labels, mais aussi de lignes, de courbes qui positionnent les parois de la grotte, et les autres caractéristiques linéaires. Pour dessiner une ligne, sélectionnez dans le menu "Edit | Insert - line" (ou Ctrl-L). Le message dans la barre de statut change en "Insert line point": en fait, une courbe est définie par plusieurs points particuliers (points angulaires, points intermédiaires, etc.). Quand vous êtes en mode insert-line, chaque point que vous cliquez devient un nouveau point de la ligne. Pour terminer la ligne, appuyez sur la touche "Echap" (et

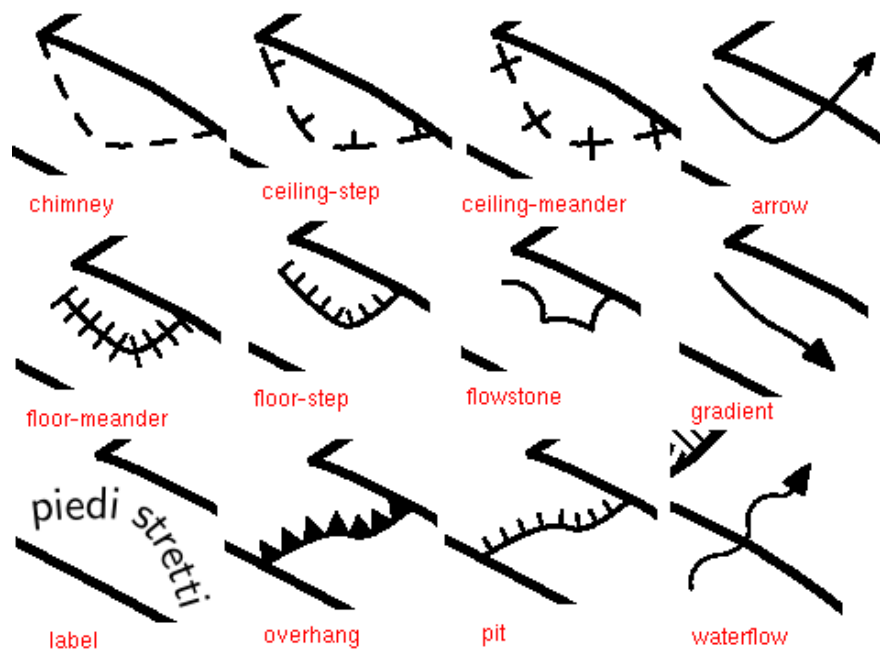
revenir ainsi en mode de sélection), ou vous pouvez presser la touche Ctrl-L et démarrer l'insertion de points pour une nouvelle ligne.

### *control point*

Vous pouvez dessiner deux types de lignes : des lignes polygonales (faites de segments droits entre deux points cliqués) et des courbes douces. Pour la première vous devez cliquer sur le sommet de la ligne polygonale. Pour la seconde, vous devez cliquer sur un point et le déplacer à la souris. Cela produit un petit segment avec le point au milieu et deux petits carrés aux extrémités. Ces carrés sont les points de contrôle de la ligne. Le jaune est la direction avant et le rouge la direction arrière. La direction du segment et sa longueur définissent comment la ligne doit passer par le point, c.à.d., sa direction et sa courbe en ce point (elle est aussi fonction du demi segment du point précédent et suivant sur la ligne). Si vous avez besoin que ce segment soit déterminé uniquement vers l'avant ou seulement vers l'arrière, vous devez désactiver la case correspondante dans le panneau "Line points" : "<<" pour désactiver la ligne vers l'arrière, et ">>" pour la désactiver vers l'avant. La case "smooth" est active quand les deux autres sont actives toutes les deux. Ainsi vous avez un contrôle complet de la courbe.

Les lignes, comme les points ont plusieurs types. En voici quelques-uns :

- "*wall*" (murs), pour les parois de la cavité;
- "*contour*" et "*border*";
- "*chimney*" (cheminée) et "*pit*" (puits);
- "*rock-border*" (bord du rocher) et "*rock-edge*" (intérieur du rocher) : le premier pour le contour du rocher, le second pour les lignes internes;
- "*section*" pour le segment d'une section (coupe de galerie);
- "*arrow*" (flèche), "*gradient*", sont des lignes terminées par des flèches
- "*water-flow*" (circulation d'eau).



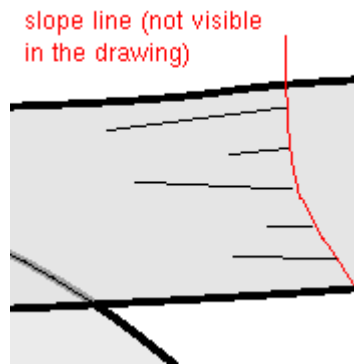
*gradient (line option)*

*contour (line type)*

Les lignes de type “contour” peuvent avoir une option “gradient”. Cela peut être “-gradient none” (rien n’est dessiné, ce qui est l’option par défaut), ou “-gradient center” (la flèche est dessinée au niveau du point). Il est également possible d’ajouter l’option “gradient point” après chaque point où vous voulez dessiner une flèche (vous devez écrire “gradient point” dans la case option du point dans l’éditeur de carte, ou vous pouvez éditer le fichier carte avec un éditeur de texte).

*slope (line type)*

Les lignes de type “slope” (ligne de pente) sont dessinées comme des segments orientés perpendiculairement à la ligne, et passant par la ligne de point. C’est pourquoi l’orientation de ces segments dépend également des points de contrôles. Au moins un des points de la ligne “slope” doit avoir un “l-size” qui définit la longueur des segments. Chaque point peut avoir une orientation qui modifie l’orientation par défaut.



*(subtype value)*

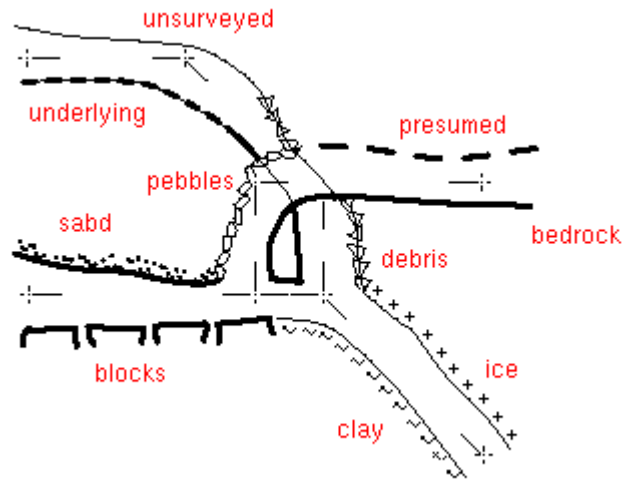
*presumed (subtype value)*

*subtype*

Certaines lignes (“wall”, “border”, “survey” and “water-flow”), comme leurs homonymes parmi les points, peuvent avoir des sous-types (option “-*subtype*”). Par exemple, la ligne de type “border” (limite, bordure) peut être “invisible”: cela est utilisé dans le cas des limites de surfaces qui ne doivent pas être dessinées sur la carte [thwiki 17]. Un autre sous-type est “presumed” (supposé): pour les lignes “wall” (parois) qui sont supposées, mais également pour les lignes de type “border”.

*wall (Types de lignes)*

Des lignes de type “wall” peuvent avoir un sous-type “bedrock” (roche), “blocks” (rocher), “clay” (argile), “debris” (débris), “ice” (glace), “sand” (sable), “pebbles” (cailloux), “unsurveyed” (non topographié), “underlying” (paroi sous-jacente), “presumed” (supposé) and “invisible” (invisible). Ceux-ci sont présentés dans le dessin ci-dessous (sauf “invisible” qui signifie de ne pas dessiner la ligne sur le plan final).



Les lignes de type “water-flow” (écoulement d’eau) peuvent avoir comme sous type “permanent”, “intermittent” et “conjectural” (supposé). Les lignes de type “border” peuvent avoir comme sous type “invisible”, “temporary”, “visible” et “presumed” (supposé).

### Orientation

Les lignes sont orientées : elles ont un petit trait jaune à leur extrémité de départ. Celui-ci doit être orienté vers l’intérieur de la galerie (où il n’y a pas le rocher). C’est pourquoi vous devez dessiner les parois de la cavité en tournant dans le sens inverse des aiguilles d’une montre ; pour les rochers, dans le sens des aiguilles d’une montre ; pour les puits et les cheminées dans le sens inverse des aiguilles. Si vous faites une erreur, vous pouvez inverser la direction de la ligne en cliquant sur le bouton “Reverse” du panneau de contrôle “Lines” de la fenêtre.

Si un point appartient à deux (ou plus) lignes, quand vous sélectionnez le point (bouton gauche de la souris) une ligne est automatiquement sélectionnée. Si ce n’est pas la ligne désirée, vous pouvez faire alterner les lignes passant par ce point en cliquant à nouveau sur ce point avec le bouton droit de la souris [thwiki 17].

Maintenant, insérez les lignes des bords de la cavité (type “wall”). Si vous faites une erreur, supprimez la ligne (Ctrl-D) et essayez à nouveau (Ctrl-L).

Il y a deux façons de dessiner une ligne fermée : soit vous terminez en cliquant sur le point de départ, ou alors vous cliquez sur la case “close” du panneau de contrôle “Line” . Si par hasard vous fermez une ligne que vous ne vouliez pas fermer, sélectionnez le dernier point de la liste dans le panneau “Line control”. Il doit avoir les mêmes coordonnées que le premier point. Avec le menu “Edit line” sélectionnez “Delete point”, et vous le supprimez ainsi de la ligne (le premier point reste sur la ligne).

Vous pouvez sélectionner chaque point individuellement sur la ligne. Une fois un point sélectionné, il peut être déplacé (glissé avec la souris). Vous devez cliquer deux fois sur le point : une première fois pour sélectionner le point et afficher un cercle rouge autour de lui. Quand vous cliquez une seconde fois sur le point (et en gardant le bouton de la souris appuyé) vous l’attrapez et le faites glisser sur le dessin. Relâchez le bouton quand le point est positionné là où vous le désirez.

Pour supprimer un point d’une ligne, allez sur le panneau “Lines” et sélectionnez le bouton “Edit line | Delete point”. Pour insérer un point dans la ligne, sélectionnez dans ce même panneau “Edit line | Insert point”. Ce bouton met *XTherion* en mode insertion de point dans une ligne (pour la ligne actuellement sélectionnée), et vous pouvez insérer des points.

Pour couper une ligne à un point précis, sélectionnez “Edit line | Split line” dans le panneau “Lines”. Si vous voulez couper la ligne entre deux points, vous devez d’abord insérer un nouveau point au niveau de la coupure. Plus de détails dans le chapitre 3.2.

### 2.6.1 Symboles et libellés

Les symboles et libelles sont associés aux points, c.à.d. à la position du point dans le dessin. Certains symboles ont une orientation. Par exemple “*water-flow*” a la direction du ruissellement de l’eau : cela active une case à cocher “orientation” et vous pouvez sélectionner une orientation en tournant la flèche jaune fixée au point, à l’aide de la souris, ou en écrivant une valeur d’angle (en degrés) dans la zone de texte, près de la case à cocher. Malheureusement, le symbole n’est pas affiché sur l’éditeur de carte, mais seulement le point est visible, et il est assez difficile de se rappeler ce à quoi chaque point correspond. Un peu d’aide est fourni par la barre de statut qui affiche le type du point quand la souris passe dessus.

Les libellés sont des points de type “label”. Le texte à afficher doit être saisi dans la case “options” comme une option “-text ...”. Vous pouvez utiliser des tags HTML pour formater le texte : `&lt;it &gt;` pour mettre en italique, `&lt;bf &gt;` pour mettre en gras, et ainsi de suite. Utilisez `&lt;br &gt;` pour faire un retour à la ligne, et `&lt;right &gt;` pour aligner à droite le texte.

### 2.6.2 Text

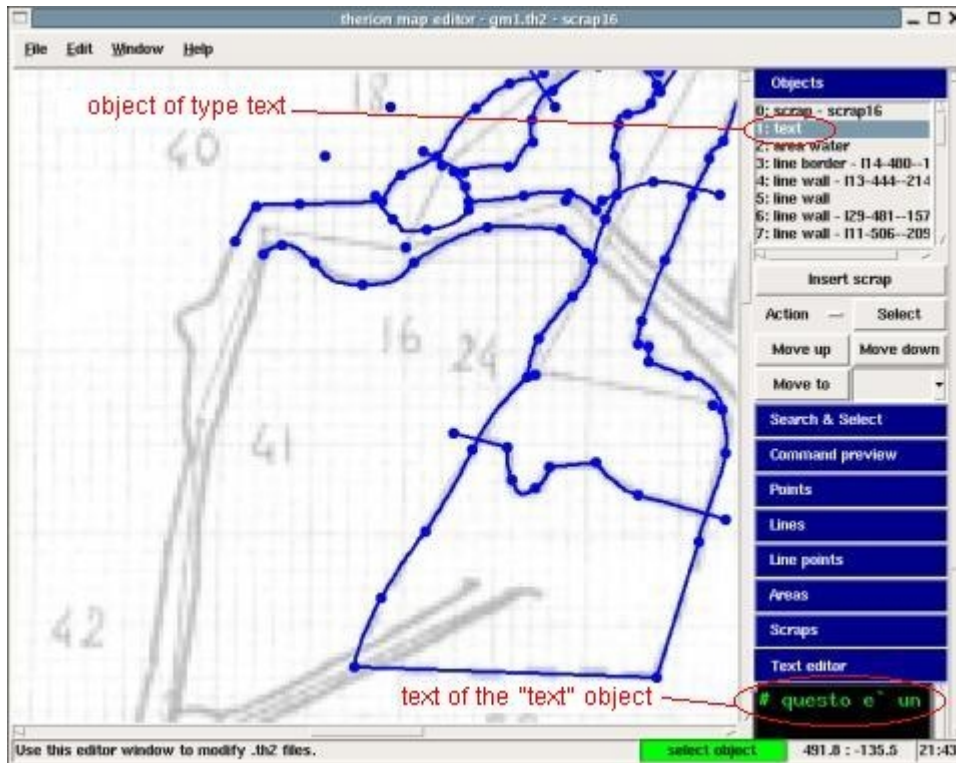
Les objets *text* sont utilisés pour écrire du texte qui doit être incluse dans le dessin final. Ils sont utilisés pour ajouter des commandes spécifiques à METAPOST ou pdfTeX (quand on a besoin de faire quelque chose de spécial).

Sélectionnez le menu “Edit | Insert - text”, ou simplement dans le panneau “Objects”, sélectionnez la commande “Action | Insert text”. Cela ouvre le panneau “Text” control, et vous pouvez entrer votre texte dans la zone de saisie noire. Quoi que vous tapiez, ce sera écrit littéralement dans le fichier de données graphiques.

Comme pour les labels, vous pouvez utiliser des tags HTML pour formater le texte. Par exemple, pour aller à la ligne, écrivez `&lt;br &gt;`*[thbook ...]*.

Maintenant, essayez d’insérer un commentaire dans vos fichiers de données.





*cross section*

### 2.6.3 Les sections transverses (coupes)

#### *section (type)*

Une section est un dessin associé à une position dans le dessin du plan ou de la coupe développée. Cette position est indiquée avec un point de type "section" et il doit avoir l'option "-scrap" suivie par le nom du scrap de la section.

#### *scale (option)*

Le scrap de la section a une autre particularité. Il n'a généralement qu'une seule station topo ; c'est pourquoi, il doit avoir une option "-scale" ainsi *Therion* peut connaître ses dimensions.

Les sections seront expliquées plus loin au chapitre 3.5.

## 2.7 Les surfaces

### *Area*

Une surface est une portion de grotte remplie avec quelque chose. Par exemple, un plan d'eau est affiché comme une surface de type "water" (eau) (ou "sump" (siphon) si c'est un siphon). D'autres objets graphiques sont des surfaces car elles remplissent des zones du dessin: "sand" (sable), "debris" (débris/cailloux), "mud" (boue), etc.

## Outline

Une surface est définie par les lignes qui font son contour. Ces lignes doivent se couper deux à deux, et être listée dans l'ordre d'apparition, dans le sens des aiguilles ou le sens inverse. Si les lignes ne se coupent pas (c.à.d. si elles ne définissent pas un contour fermé, le résultat est imprévisible. Quand vous dessinez des lignes de bordure d'une surface (type "border"), qu'elles soient visibles ou invisibles (sous type "invisible"), vous devez les faire légèrement dépasser des parois de la grotte pour être sûr que les lignes se coupent. En effet, *Therion* ne dessine pas les portions de lignes qui dépassent hors des traits des limites de la grotte (qui sont généralement définis par les parois de la grotte).

Pour insérer une nouvelle surface, c.à.d. pour passer en mode insertion de surface, sélectionnez le menu "Edit | Insert - area", ou le raccourci clavier Ctrl-A. Chaque ligne sur laquelle vous cliquez est ajoutée à la liste des lignes qui définissent le contour de la surface. L'insertion se termine quand vous appuyez sur la touche "Echap" (pour revenir au mode de sélection), ou Ctrl-A pour insérer une nouvelle surface.

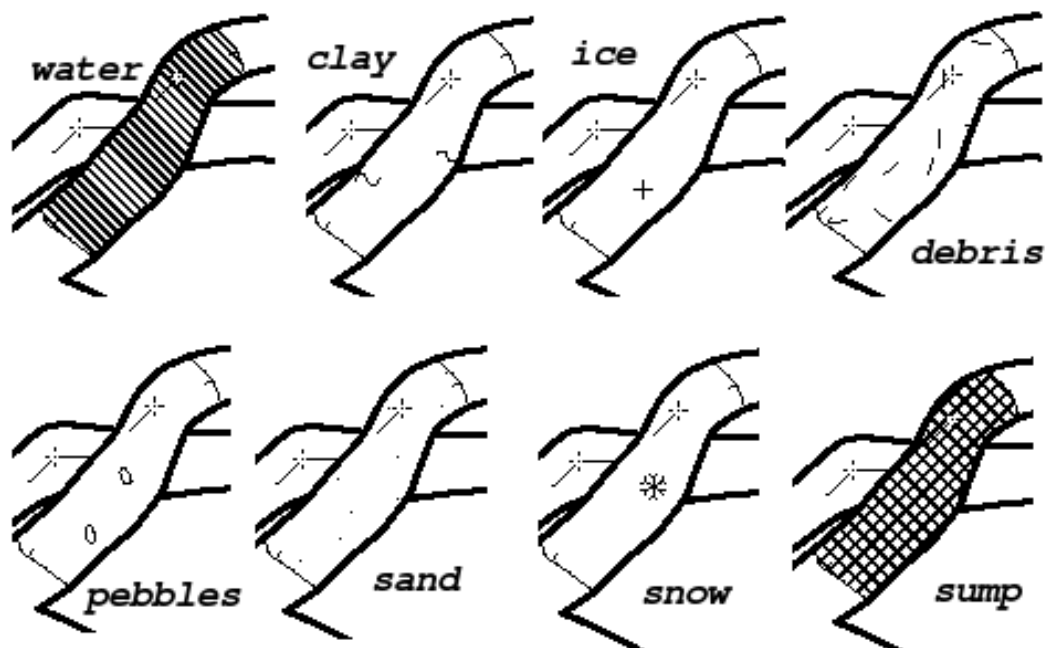
Faites attention de ne pas cliquer deux fois sur la même ligne, sinon elle apparaît deux fois dans la liste des contours de la surface. Vous pouvez le voir dans le panneau "Areas" qui affiche la liste des lignes formant le contour. Si une ligne est répétée deux fois (par erreur), vous devez en supprimer une afin qu'elle n'apparaisse qu'une fois.

Vous pouvez supprimer une ligne du contour d'une surface. Sélectionnez la ligne et appuyez sur le bouton "Delete" dans le panneau "Areas". La ligne est supprimée de la liste des lignes bordant la surface, mais elle n'est pas supprimée du dessin.

La seule solution pour sélectionner une surface est de le faire à travers le panneau "Objects", en cliquant sur la ligne correspondant à la surface. Vous pouvez observer que la ligne des contours de la surface apparaît sur la fenêtre de dessin.

La commande "area" a comme argument, uniquement le type de la surface. Ce type peut être un des suivants :

- "*water*", flaqué l'eau
- "*sump*" (siphon)
- "*sand*" (sable)
- "*debris*" (débris/cailloux)
- "*blocks*" (rochers)
- "*snow*" (neige)
- "*ice*" (glace)
- "*clay*" (argile)
- "*pebbles*" (galets)



*place* (option)

*clip* (option)

*visibility* (option)

Les options de la commande “area” sont

- “*place*” indique le niveau auquel on doit dessiner la surface (par rapport aux autres éléments du dessin). Cela peut être “bottom” (en dessous), “top” (au dessus), ou “default” (par défaut : même valeur que par défaut).
- “*clip*” (découpe) indique si l’objet (de la surface) doit être inclus (et découpé pour être) dans la limite intérieure de la grotte ou non. L’option peut avoir les valeurs “on”(inclus) ou “off” (non limité à la grotte).
- “*visibility*” (visible) peut être “on” (vrai) ou “off” (faux).
- “*context*” affiche le dessin de la surface avec d’autres symboles (voir option symbol-hide/symbol-show).

*context* (option)

L’exemple ci-dessous est une surface de type “water” (eau) bordée de lignes de type “wall” et “border”. Une des lignes “border”, dans la partie inférieure, a une option “-visibility off”. La surface a un contexte “-context line wall:debris”. L’affichage de cette surface est dessiné avec les symboles “debris” sur son contour, et la surface “water” n’est pas dessinée. Pour dessiner la surface “water” vous pouvez ajouter une seconde surface avec le même contour, mais sans “context”. Si la surface avec le “context” est au-dessus l’autre dans la liste du panneau des “Objects”, elle sera dessinée par dessus l’autre surface, et les triangles des “debris” seront visibles le long du bord de l’eau et sur la paroi de la cavité. Si elle est en dessous, elle sera *couverte* par (l’eau) “water”, et les triangles des “debris” ne seront visibles qu’en limite de la surface du plan d’eau.



Pour les lignes, l'option "context" semble changer le type de la ligne. Par exemple la ligne de type *border* avec une option "-context line overhang" est dessinée comme une ligne de type *overhang* (ligne de surplomb).

La même chose arrive pour les points. Un point de type "moonmilk" avec "-context point entrance" est dessiné comme un symbole *entrance*. Il y a des exceptions : un point de type "label" est toujours dessiné avec le texte du label.

## 2.8 Syntaxe du *thconfig*

---

*thconfig* (Fichier de configuration de *Therion*)

Les fichiers de configuration [ThBook 38] sont des fichiers texte avec des commentaires (commençant par '#'), et des lignes de commandes. Les commandes peuvent être sur une ou plusieurs lignes. Dans ce cas, il y a une terminaison de la commande qui est le nom de la commande préfixé par "end". Les commandes sur une seule ligne peuvent être continuées sur plusieurs lignes en terminant la ligne par le continueur de séquence (antislash '\')suivit de retour chariot).

*encoding* (commande *Therion*)

*encoding*. Spécifie le mode d'encodage (codes ascii) des données. Exemples: UTF-8, ASCII, ISO-8859-1, ...

*source* (commande *thconfig*)

*source*. Elle est utilisée pour indiquer un fichier de donnée qui doit être utilisé dans les calculs par *Therion*.

*input* (commande *Therion*)

*input*. Est utilisé pour inclure un fichier, comme pour la commande "#include" du préprocesseur C. Elle ne peut pas être utilisée dans un bloc `layout- endlayout`.

*select* (commande *thconfig*)

*export*. Cette commande sélectionne un objet (soit un *survey* ou une *map*) à exporter dans un fichier. Les règles de sélection nous permettent d'organiser les cartes et les scraps avec polyvalence [thbook 38].

1. Par défaut, il n'y a pas de commande "select" avec un *survey*, tous les fichiers topographiques sont sélectionnés pour l'export.
2. Si seul les fichiers topographiques sont sélectionnés, et qu'il n'y a pas de commande "select" pour une carte, toutes les cartes des fichiers topographiques sélectionnés sont sélectionnées.
3. Si des fichiers topographiques sont sélectionnés et qu'il n'y a pas de définition de carte dans le fichier topographiques, toutes les centerlines du fichier topographique sont exportés.
4. s'il y a des cartes sélectionnées, elles sont exportées.

La commande "select" a des options,

- *recursive*, s'applique à un fichier topo "sélectionné" (select): elle définit que tous les sous-fichiers topographiques du fichier topo sont sélectionnés, récursivement.
- *map-level N*, ne s'applique que sur la carte "select": elle indique le niveau où stopper l'exportation des cartes lors de la génération de l'atlas.
- *chapter-level N*, ne s'applique que sur la carte "select": elle définit le chapter-level (niveau de chapitre) où stopper l'exportation des cartes lors de la génération de l'atlas.

*unselect* (commande *thconfig*)

*unselect*. C'est l'opposé de *select*. Elle est utilisée quand vous voulez inclure un objet dans un export, et que vous ne voulez pas l'inclure dans les exports suivants. Vous mettez un "unselect" de l'objet entre les commandes export.

*layout* (commande *thconfig*)

*layout*. Elle indique un ensemble d'options d'affichage 2D (de la carte générée). Elle prend comme argument un id (nom) du layout, qui doit être utilisé plus loin pour le référencer. Elle a plusieurs options. Parmi celles-ci :

- *copy Layout\_id* : copie les options d'un autre layout dans celui-ci;
- *scale from to* : définit l'échelle de la carte. Par exemple "scale 1 500". Par défaut elle est de 1:200.
- *base-scale from to* : recalcule d'échelle de la carte par un facteur *scale/base\_scale*;
- *units* : a comme argument "metric" (par défaut) ou "imperial";
- *rotate degrees* : fait faire une rotation au dessin sur la carte finale.
- *symbol-set* : Indique le jeu de symboles à utiliser: UIS (Union Internationale de Spéléologie), ASF (Australian Speleological Federation), CCNP (Carlsbad Caverns National Park) or SKBB (Speleoklub Banska Bystrica); par exemple "symbol-set UIS". A ce jour, les différents symboles diffèrent peut.
- *symbol-hide* : indique des symboles ou des groupes de symboles qui ne doivent pas être dessinés. Par exemple: "symbol-hide group cave-centerline", "symbol-hide point cave-station" ... Il y a aussi la commande *symbol-show*.
- *size width height units* : indique les dimensions (pour la carte, l'option *page-grid* doit être "on"). Exemple: "size 18 22.2 cm"
- *overlap value units* : recouvrement entre les cartes

- *grid* a pour argument “off” (par défaut), “top” ou “bottom”
- *grid-origin* : indique les coordonnées de la grille d'origine. Exemple: “grid-origin 0 0 0 m”
- *grid-size* : définit l'espacement de la grille (par défaut 10 m). Exemple: “grid-size 50 m”
- *transparency* : a pour argument “on” ou “off”, et définit and indique si une galerie de la grotte qui passe sous un autre doit être visible ou non (par défaut “on”), et elle doit être utilisé avec l'option *opacity*)
- *opacity* : définit la valeur de l'opacité pour la galerie inférieure, valeur entre 0 et 100, (par défaut 70)

Les options suivantes s'appliquent aux atlas:

- *page-setup* définit la taille de la page. Son argument est la valeur numérique suivie de l'unité. Par exemple, “page-setup 29.7 cm”
- *page-numbers* a comme argument “on” (par défaut) ou “off”
- *exclude-pages* a comme argument “on” ou “off”, et la liste des numéros de page à exclure (les numéros de pages débutent à 1). Par exemple, “exclude-pages on 2,3,6-8,19”
- *title-page* a comme argument “on” (par défaut) ou “off”. Il fait référence au titre de la page en face du chapitre.
- *nav-factor* définit le facteur de zoom de la grille de navigation. Par défaut il est de 30.

*legend* (option layout)

Options *legend*:

- *legend* a comme argument “off”, “on” ou “all”
- *scale-bar*. Par exemple “scale-bar 10 m”. Le pas de la grille est ajusté à la scale-bar si elle n'est pas spécifiée par l'option *grid-size*.
- *map-header* a trois arguments: X, Y, et un parmi “off”, “center”, “n”, “e”, ..., “sw”. Par exemple, “map-header 0 0 n”
- *statistics* ajoute quelques statistiques. Elle a comme argument “explo-length” (longueurs d'explo) ou “topo-length” (longueur de topo), elle peut avoir comme valeur “on” ou “off”. Le premier permet d'écrire les statistiques, le second de ne pas les écrire. Quand elle a comme argument “explo”, “topo”, “carto” ou “copyright” la valeur peut être “all” (on écrit tous les noms), “off” (on écrit rien), ou un nombre N (on écrit les N premiers noms).
- *language* a comme argument le langage à utiliser pour la carte générée.
- *map-comment* est suivit par la chaîne de caractères du commentaire que vous voulez inclure sur la carte. Le commentaire est placé entre le dessin et la légende.
- *debug* peut être “on”, “all”, “first”, “second”, “scrap-names” ou “off”

Il y a aussi des options spécifiques aux PDF. Pour leur nom c'est assez parlant. Les options affectent des valeurs aux propriétés des documents PDF, et elles sont suivies par la valeur de l'option (chaîne de caractères entre “double quotes” si elle a des espaces).

- *doc-author*
- *doc-keyword*
- *doc-subject*
- *doc-title*

*export* (commande *thconfig*)

*export*. Définit le fichier qui doit être créé. Elle a comme argument le type de fichier, “map”, “atlas”, “model” ou “*database*”. Parmi les options, il y a,

- *output* indique le nom du fichier. Par défaut “cave.XXX” où l’extension “XXX” varie suivant le format de la sortie (type de la carte).
- *fmt* spécifie le format de la sortie. Les formats possibles dépendent du type de la sortie:
- *maps* peut être “pdf”, “svg”, “xvi”.
- *atlases* uniquement “pdf”
- *models* peut être “*Therion*” (par défaut le format natif), “compass” (plt), “survex” (3d), “dxf”, “vrml”, et “3dmf”
- *databases* uniquement “sql”
- *enable* et *disable*, (uniquement pour les modèles) sélectionne quels sont les éléments à exporter sur la carte: “wall”, “centerline”, “surface”, “all”;
- *projection ID* (pour les cartes et les atlas) indique le type de projection, comme pour les scraps. Cela peut être “none” (cross-sections), “plan” (projection horizontale), “elevation” (projection verticale, également possible avec une projection sur un axe en degrés), and “extended”.
- *layout ID* (pour les cartes et les atlas) inclus un layout définit précédemment.
- *encoding* (pour les bases de données) indique le mode d’encodage.
- Il est également possible de définir des options de layout en les faisant précéder du préfixe “layout-”.

## 2.9 Syntaxe des données

---

La syntaxe de *Therion* est détaillée dans le manuel de référence “*Therion book*” [thbook 10-11].

Les fichiers de données *Therion* sont des fichiers texte, avec des lignes de commandes (ayant des arguments et des options), des lignes de données, et des lignes de commentaires (ces lignes commencent avec le caractère ‘#’). Les commandes peuvent être sur une seule ligne ou sur plusieurs lignes. Dans ce cas, il y a un mot de terminaison de la commande, qui est composé du nom de la commande préfixé par “end”. Par exemple, la commande *centerline* est terminée par l’instruction *endcenterline*.

Les lignes de données peuvent contenir des valeurs numériques ou des options pour la commande. Dans le cas des options, cela s’applique aux données suivant l’option.

### Couleurs

Le caractère ‘#’ indique le début d’un commentaire. Les commentaires sur une ligne de donnée sont autorisés: tout ce qui suit un ‘#’ jusqu’à la fin de la ligne est considéré comme un commentaire (et ignoré). Le caractère backslash ‘\’ à la fin de la ligne indique que celle-ci se poursuit à la ligne suivante s’il s’agissait d’une commande sur une seule ligne. Les arguments et les options ayant des espaces doivent être encadrés par des doubles quotes ‘”’, s’ils sont des chaînes de caractères, ou des crochets ‘[ ]’, pour les valeurs numériques.

Par exemple, la valeur RGB d'une couleur est composée de trois nombres (entre 0 et 100) et elle doit être écrite entre crochets : par exemple [100 0 0] est la couleur rouge.

Les mots clés peuvent contenir les lettres de l'alphabet, des chiffres, le souligner '\_', le signe moins '-', et le slash '/'. Les mots peuvent contenir également le signe plus '+', l'étoile '\*', la virgule ',', le point '.', et les guillemets"". Pour utiliser un mot clé comme valeur dans une ligne de donnée, vous devez le préfixer avec un point d'exclamation '!'.  
*input*

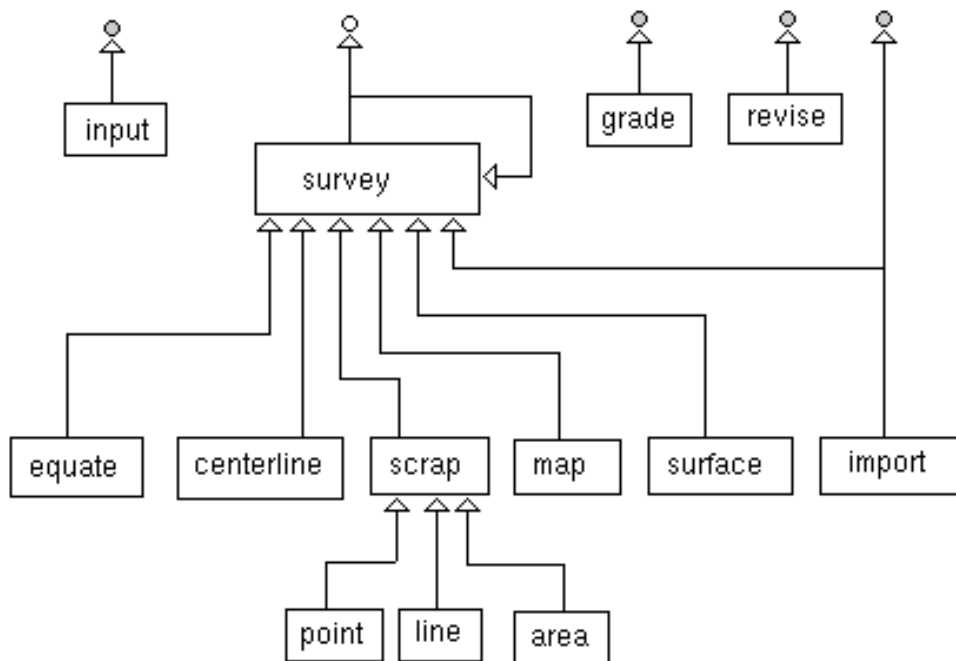
### Date

Le format des dates est aaaa.mm.jj\$hh:mm:ss.ss. Un intervalle de date est composé par deux dates jointes par un trait (signe moins '-'). Le nom d'une personne est composé de deux chaînes séparée par un espace, c.à.d., "nom prénom". Seul un espace est autorisé. Parmi les unités de mesure, il y a le *meter* (le mètre) *centimeter* (le centimètre) (également *m* et *cm*), *degree* (*deg*) (degrés) et *percent* (pour-cent %).

### 2.B.1 Les commandes de *Therion*

#### Contextes

Les commandes de *Therion* ont un contexte, c.à.d., une commande doit se placer dans une autre qui est son contexte. La figure suivante montre la hiérarchie des contextes : les contextes des commandes sont indiqués par des flèches.



#### *encoding* (commande *Therion*)

*encoding tipo*. Cette commande définit l'encodage de table de caractères utilisé pour écrire le fichier de données. Par exemple *encoding UTF-8* ou *encoding ASCII*.

#### *input* (*Therion* command)



*input* file. Cette commande indique à *Therion* de lire un autre fichier. Si le nom du fichier contient un répertoire, vous devez utiliser le slash '/' pour une meilleur portabilité. Il est conseillé d'éviter d'utiliser les noms en chemin absolu.

*survey* (commande de fichiers .th )

*survey name*. Cette commande définit un niveau dans le domaine des fichiers topographiques. Le nom du niveau, est un argument de la commande (le nom de la topographie). Parmi les options, il y a *declination* (pour indiquer la valeur de la correction de la déclinaison magnétique que l'on doit appliquer aux données) et *title*, qui indique le titre de la topographie.

*centerline* (commande de fichiers .th)

*centerline*. Cette commande définit le début d'un fichier de données topographiques. Elle a plusieurs options :

- *date valeur\_date* : date de la séance de topographie
- *team personnes [roles]* membres de l'équipe de topographie et leurs rôles (*station, length, tape, compass, bearing, clino, gradient, counter, depth, station, position, notes, pictures, pics, instruments (insts), assistant (dog).*).
- *units quantity [factor] units* : unités de mesure pour une donnée (quantity prend pour valeur length, clino, compass). Un facteur d'échelle peut être spécifié ex: "units length 0.01 meters" pour des longueurs en centimètres.
- *sd quantity value units* : déviation standard pour une unité de mesure (quantity= compass ou clino)
- *declination value units* : déclinaison magnétique
- *infer what on/off*, ex : "*infer plumb on*" indique que les visées +/- 90 ne doivent pas être corrigées de la correction de l'inclinaison (correction du clinomètre). "*infer equates on*" indique que les visées de longueur 0 doivent être considérées comme une commande "*equate*" de points topo (pas de correction des longueurs.
- *mark [stations] type* : ici le type de point topo (paramètre type) peut être *fixed, painted, ou temporary*
- *station "station comment"* : associe un commentaire à la station. Si le commentaire a des espaces, celui-ci doit être encadré par des doubles quotes "". Exemple: *station 32 "continuable, with less rock"*.
- *fix station x y z* : fixe les coordonnées géographiques d'un point topo. Exemple "fix 1 1529802.9 5089076.7 2214.6" définit les coordonnées (dans ce cas le système de référence est Roma-40) de la station 1: la coordonnée X la coordonnée vers l'Est, Y vers le Nord, et Z est l'altitude. Elles sont toutes indiquées en mètres.
- *Flag* : "flag surface" indique que les visées topo suivantes correspondent à une topographie de surface et ne doivent pas être prises en compte dans les statistiques de la cavité. "flag duplicate" indique que les visées sont dupliquées et qu'il ne faut pas en tenir compte dans le calcul du développement de la cavité.
- *equate stations* : définit l'égalité de deux stations topos (deux noms pour le même point lors de 2 séances différentes). Par exemple "equate 1 24@main\_passage".
- *data style data\_order* : indique le type et l'ordre des données topographiques sur les lignes de données topographiques (longueur, azimuth, pente, ... )
- *walls type* : le type peut être *auto, on, or off*. Indique si les données LRUD (largeurs gauche droit, haut, bas) doivent être utilisées pour générer le modèle 3D.

- *extend type [stations]* : Est utilisé pour la coupe développée. Ce type peut être *left, right, break, start, ignore*

### *scrap*

*scrap*. Un scrap contient une portion de la topographie de la cavité sans passages superposés. Il couvre en général une centaine de mètres de galerie. Il contient des points (symboles et textes), des lignes (lines) et des surfaces (areas). Le contour du scrap est composé de lignes avec l'option "*-outline*". Elles peuvent être "out", "in" ou "none". Les lignes de type "*wall*" ont *-outline* out par défaut. D'autres lignes ont *-outline* none par défaut. Les lignes de contour ne doivent pas se couper, sinon, METAPOST donnera le message d'erreur (warning) (*Therion* voit le message mais continue). C'est une bonne méthode que de vérifier ces messages et de corriger les problèmes systématiquement.

Un *scrap* doit avoir deux stations topographiques qui lui donnent une référence de positionnement sur la topographie. Si le scrap n'a qu'une seule station (ou aucune), comme c'est le cas dans les sections de galerie, il doit avoir l'option *-scale*. Les options de *scrap* sont :

- *projection* indique le type de projection. Cela peut être none (section de galerie), plan, elevation (projection verticale), and extended (coupe développée).
- *scale* pour une mise à l'échelle à la main en indiquant par une flèche rouge quelle est la dimension et l'orientation de la section.
- *stations* ... liste des stations topo qui seront dessinées dans le scrap et qui ne sont pas listées parmi les points.
- *walls*. Elle peut être "on", "off" ou "auto". Elle indique si le scrap doit être utilisé ou non dans la construction 3D de la cavité.
- ...
- *author* indique l'auteur du scrap;
- *title* indique le titre du scrap;
- *copyright* indique le copyright pour ce scrap.

### *point* (commande de fichiers .th2)

*point x y tipo*. Cette commande définit un point dans le dessin, ses coordonnées et son type. La plus part des symboles topographiques sont associés à des types de points. Les libellés de texte sont aussi des objets "point". Il y a également des points qui n'ont pas de représentation dans le dessin (par exemple des points qui définissent la dimension des galeries). Les types de points sont :

- *objects* (objets):

### *station* (type)

- "*station*"; station topographique. Elle doit avoir l'option "*-name*" avec le nom d'une station (ex: a1@topo1 où a1 est le nom d'un point topo dans le fichier topo1.th) *section* (type)
- "*section*" est une référence pour une section de galerie, elle spécifie le nom du scrap contenant la section avec l'option "*-scrap*"
- "*dimensions*" indique la dimension de la galerie. Il doit avoir l'option "*-value*"

### *labels* (libellés texte)

- “*label*”, libellé, le text est précisé dans l'option *-text* et “mis entre quotes”
- “*remark*”, une note, remarque
- “*altitude*”, pour indiquer l'altitude du point par rapport à la grille.
- “*height*” (hauteur) est utilisé pour la hauteur d'éléments de la grotte comme les cheminées.
- “*passage-height*” (hauteur de passage) pour indiquer la hauteur des galeries;
- “*station-name*”, (nom de la station) pour indiquer le nom de la station sur la carte.
- “*date*”, pour indiquer la date
- remplissages, surfaces: “*bedrock*” (rocher compacte), *sand* (sable), *raft* (calcite flottante), *clay* (argile), *pebbles* (galets), *debris* (débris, cailloutis), *blocks* (blocs), *water* (eau), *ice* (glace), *guano* (guano), *snow* (neige)
- spéléothèmes: *flowstone* (coulées de calcite), *moonmilk* (mondmilch), stalactite, stalagmite, *pillar* (pilier), *curtain* (draperie), *helictite* (excentrique/hélictite), *soda-straw* (fistuleuses), crystal (cristaux), *wall-calcite* (paroi de calcite), *popcorn* (choux-fleur), *disk* (disque), *gypsum* (gypse), *gypsum-flower* (fleur de gypse), aragonite, *cave-pearl* (perle des cavernes), *rimstone-pool* (gours), *rimstone-dam* (bordure de gour), *anastomosis* (anastomose), *karren* (lappiez), *scallops* (coups de gouge), *flute* (flûte ?), *raft-cone* (cone)
- équipement: *anchor* (ancrage), *rope* (corde), *fixed-ladder* (échelle fixe), *steps* (marche), *bridge* (pont), *traverse* (traversée), *camp* (camps), *no-equipement* (pas d'équipement)
- entrée et terminaison: continuation, *narrow-end* (passage impénétrable), *low-end* (passage bas), *flowstone-choke* (trémie calcifiée), *breakdown-choke* (trémie), *entrance* (entrée);
- autres: *archo-material* (matériel archéologique), paleo-material (matériel paléolithique), *vegetable-debris* (débris végétaux), *root* (racines), *water-flow* (circulation d'eau), *spring* (source d'eau), *sink* (perte d'eau), *air-draught* (courant d'air, l'intensité est indiqué par l'option “-scale”), *gradient* (pente)

*place* (option)

*clip* (option)

*context* (option contexte)

Certains types de points peuvent avoir plusieurs options et *subtype* :

- “*station*” peut avoir un sous-type (subtype) “temporary” (temporaire), “painted” (peint), “natural” (naturel) and “fixed” (fixe);
- “*water-flow*” peut avoir un sous-type (subtype) “permanent”, “intermittent” and “paleo” (paléo)
- *-orientation*, la valeur de l'orientation de la flèche est donnée en degrés
- *-align*: généralement pour le texte. Il peut être “center” (centré), “top” (au sommet), “left” (à gauche), “right” (à droite), etc.
- *-scale* indique la dimension du symbole de très petit (xs) à très large (xl) : “xs”, “s”, “m”, “l” ou “xl” [“tiny” (très petit), “small” (petit), “normal” (normal), “large” (large), “huge” (très large)]. À un niveau supérieur (des fichiers *Therion*), les dimensions peuvent être fixées avec l'option “*base-scale*” (ou avec la commande “*u:=dimension*” dans une section “code metapost”, ex: “*u:=10pt*”, “*length:=dimension*” pour les lignes).

- *-place*, indique le niveau où positionner le symbole durant la construction de la carte. Par exemple *"-place bottom"* (permet de spécifier l'ordre d'affichage des symboles, et donc l'ordre de superposition des dessins).
- *-clip*, indique si le symbole est clippé ou non (découpé et non affiché pour toute la partie sortant de la grotte, en dehors des murs).
- *-visibility*, si elle est *"off"* le symbole n'est pas dessiné
- *-context*, définit le contexte du symbole, c.à.d. le symbole courant est dessiné de la même façon que le contexte courant sur la carte.
- *-id*, indique le code du symbole (unique dans tout les fichiers *Therion*).
- *-name*, pour les stations topo "station" indique le nom de la station (a1@topo2).
- *-extend*,
- *-scrap*, nom du scrap à affiché en ce point (dessin de section de galerie)
- *-text*, texte à afficher sur la carte (libellé)
- *-value*, pour certains types de points, valeur caractérisant le point.

*line* (fichier de commande .th2)

*type de ligne*. Les commandes *"line"*, comme les commandes *"point"*, ont un type : *wall* (paroi), *contour* (courbe de niveau), *slope* (pente), ou peuvent définir un remplissage, une étiquette (*label*), ou bien encore un type spécial de lignes (*border* (bordure), *arrow* (flèche), *section* (), *survey* (topographie)).

La différence entre *"gradient"* (gradient de pente) et *"slope"* (pente) est que la première dessine une simple flèche (dans la direction de la ligne), alors que la seconde dessine plusieurs traits (dans la direction orthogonale de la ligne tracée sur la carte vectorielle [thmail 2006-05-05]).

*visibility* (option)

La commande *"line"* peut avoir plusieurs options :

*-subtype*. Les sous types suivants sont possibles :

- Pour les lignes de type *"wall"* (paroi): invisible, *bedrock* (roche mère), *sand* (sable), *clay* (argile), *pebbles* (galets), *debris* (débris), *blocks* (rochers), *ice* (glace), *underlying* (paroi sous-jacente), *presumed* (supposé);
- Pour les lignes de type *"border"* (bordure): visible, invisible, *temporary* (temporaire), *presumed* (supposé);
- Pour les lignes de type *"water-flow"* (circulation d'eau): permanent, *conjectural* (supposée), *intermittent* (temporaire)
- Pour les lignes de type *"survey"*: *cave* (grotte), surface,
- *-close* l'option peut être *"on"* ou *"off"* (fermée ou non) et elle indique si la ligne est fermée (boucle) ou non.
- *-mark*, prend en paramètre un nom, est utilisé pour marquer un point de la liste (voir la commande *join* de *Therion*)
- *-orientation*, prend en paramètre un angle en degré. indique l'orientation des symboles le long de la ligne (lorsque la ligne est associée à des symboles comme argile, glace, neige, ...)

- **-outline**, indique si ligne fait partie des contours du scrap (dessin) (sa valeur est "in" (dedans), "out" (bordure de la cavité), ou "none"). Par défaut, les lignes de type "wall" (paroi) ont l'option "-outline out" et les autres lignes n'ont pas d'option "outline" [thwiki 9]. Utilisez l'option "-outline in" pour les piliers à l'intérieur de la cavité.
- **-reverse**, indique que la direction de la ligne est inversée (inversion du coté intérieur/extérieur de la ligne);
- **-size**, indique l'épaisseur de la ligne (valeur du paramètre en ? (pixels?));
- **-smooth**, ("on", "off", "auto") indique si la ligne a des contours doux en ce point, la valeur par défaut est "auto"
- **-adjust**, ("horizontal", "vertical") est utilisé pour l'épaisseur des lignes horizontales et/ou verticales: Le point qui a cette option est aligné avec le point précédent, respectivement de façon horizontale, ou verticale. Cette option n'a des effets que pour les coupes développées, par pour les plans.
- **-place** indique à quel niveau positionner la ligne quand la carte est générée. Elle peut avoir comme valeur "default" (défaut), "bottom" (bas), ou "top" (haut).
- **-clip** peut être "on" (oui) ou "off" (non). Indique si le symbole doit être découpé (clippé) dans le tracé de la grotte ou non);
- **-visibility** indique si la ligne doit être dessinée ou non. Elle peut avoir comme valeur "on" (par défaut) ou "off" (non dessinée).
- **context** indique le contexte de la ligne, c.à.d. quel type de trait doit être utilisé pour dessiner la ligne. La commande prend 2 paramètres "point", ou "line", ou "area" suivit du type de symbole.
- **-altitude**, ne peut être indiqué que sur les lignes de type "wall".
- **-border**, ligne de type bordure (pour les surfaces : area)
- **-gradient**, ligne de gradient de pente. Prend un paramètre qui indique où positionner le symbole de gradient : "none" (aucun), "center" (centre de la ligne), "point" (au niveau du point), s'il n'y a aucune spécification, le symbole sera positionné en fonction du jeu de symboles UIS utilisé.
- **-head**, n'est utilisé qu'avec des lignes de type "arrow" (flèche), indique où doit se trouver la tête de la flèche. Prend un paramètre qui indique "begin" (début), "end" (fin), "both" (les deux), "none" (aucun). Par défaut, c'est la valeur "end" (fin) qui s'applique.
- **-text**. Pour les lignes de type "label". Contient le texte à afficher dans l'étiquette (en paramètre).

## Area

**area**. Une surface est définie par les lignes qui forment son contour. Ces lignes peuvent avoir n'importe quel type, mais elles doivent être indiquées dans l'ordre et toutes se couper consécutivement. Le moyen le plus simple pour définir une surface est de tracer une ligne et d'activer pour cette ligne l'option "**close**". Mais ceci n'est pas toujours possible.

La commande "**area**" a un type (le type de la surface):

- **water**, une vasque d'eau
- **sump**, un siphon
- **sand**, du sable
- **debris**, des débris
- **blocks**, des blocks
- **snow**, de la neige
- **ice**, de la glace
- **clay**, de l'argile
- **pebbles**. des galets

La commande “*area*” peut avoir des options:

- *place* indique à quel niveau (de dessin) placer la surface lors du tracé de la carte (pour qu'une surface superpose une autre surface). Paramètres “*bottom*” (dessous), “*default*” (par défaut), “*top*” (dessus). Par exemple, “*-place top*”.
- *clip* peut être “on” or “off”. Cette option est importante quand vous devez joindre deux scraps (dessins) et que vous avez besoin de continuer la surface en dehors de la limite de ce scrap : ajoutez l'option “*-clip off*”.
- *visibility* peut être “on” ou “off”. Si elle est “off” la surface n'est pas dessinée.
- *context* indique comment dessiner la surface, c.à.d. quel type de surface utiliser pour le dessin. Prend deux paramètres : “point” (point), “line” (ligne), “area” (surface) et le second : le type de symbole (voir option “type”). Elle doit être utilisée avec les options “*symbol-hide*”/“*symbol-show*” (masquer/afficher les symboles). Les symboles seront affichés ou masqués suivant les règles spécifiques du type de symbole indiqué en second paramètre.

*join* (fichier de commande .th)

*join*. La commande “join” connecte deux scraps, ou un ensemble de deux ou plusieurs points de la carte. Il est conseillé de mettre en place la commande “join” dans une grotte simple (une unique galerie), sinon, vous devrez mettre des commandes joins pour chaque objet. Il y a deux syntaxes de “join” :

```
join point1 point2 ... pointN
```

```
join scrap1 scrap2
```

Dans le premier cas, vous associez les points. Le nom du point utilisé dans la commande “join” doit être l'id du point dans le scrap (commande “point”) ou line-point (point appartenant à une commande line). Dans ce dernier cas, l'id est l'id de la ligne suivit par l'index du point; par exemple “my\_line:2” indique le troisième point de la ligne “my\_line” (l'indice des points début à 0). “my\_line:end” est les dernier point de la ligne “my\_line”.

Des exemples de “join” sont disponibles dans le chapitre 3.

*wall* (type de ligne)

Dans le second cas de syntaxe, vous joignez deux scraps. Seul les lignes de type “wall” sont jointes. Pour continuer un élément graphique entre les lignes du scrap (ou il est définit), il doit avoir l'option “*-clip off*” (par exemple pour une surface qui s'étend entre deux scraps).

*join* a deux options:

- *smooth* peut prendre le paramètre “on” ou “off”. Il indique si les lignes de jonction entre les scraps doivent être courbe (“on”) ou droites (“off”).
- *count*, qui est suivit par un chiffre, indique le nombre de connexions pour la jonction des scraps.

*equate* (fichier de commande .th)

*equate*. Identifie deux (ou plus) noms de stations. Indique que ces différents noms représentent le même point physique (stations topos équivalentes). Voir le fichier topographique pour plus de détails.

*map* (fichier de commande .th)

*break* (fichier de commande .th)

**map**. Une carte est une collection de scraps ou de cartes (**maps**) (vous ne pouvez pas les mélanger entre elles). Elles peuvent inclure un fichier topographique (**survey**) (la centerline sera affichée avec la carte) Cette commande indique comment assembler les scraps (ou cartes) et les fichiers topographiques. La commande “**map**” est multilignes,

```
map id [opzioni]
  &lt;ids of scraps or maps&gt;      &lt;--- level 0
  break
  &lt;ids of scraps or maps&gt;      &lt;--- level 1
  preview &lt;above|below&gt; another_map
endmap
```

La commande **break** découpe la carte en plusieurs niveaux (plusieurs étages de galerie). Il n'est pas possible de mélanger entre eux les id de scraps et les id de cartes: seul un type doit être utilisé. La carte est dessinée en débutant par le niveau le plus bas, c'est pourquoi les scraps de niveau “1” sont couverts par ceux de niveau “0”. La commande “**preview**” incluse les tracés des autres cartes.

La commande **map** a des options:

- **title** indique le titre de la carte;
- **proj** indique le type de projection type. Il peut être “**none**” (sections), “**plan**”, “**extended**” (coupes développées), et “**elevation**” (projection verticale).

*surface* (fichier de command .th)

**surface**. C'est une commande multiligne, utilisée pour inclure des données topographiques de surface. Elle peut être fournie comme un fichier image, ou comme une grille d'altitude. C'est pourquoi il y a deux syntaxes possibles. Dans le premier cas, vous devez indiquer le nom du fichier et sa relation à la topographie,

```
surface
  bitmap file_name calibration
endsurface
```

où la “**calibration**” associe deux points dans le fichier (indiqué comme une paire X-Y en pixel avec l'origine (0,0) dans le coin en bas à gauche), à deux points dans la topographie, indiqués en x-y en mètres, ou comme des noms de stations topographiques. C'est pourquoi la syntaxe est

```
X1 Y1 x1 y1 X2 Y2 x2 y2
\br
X1 Y1 station_1 X2 Y2 station_2
```

Dans le second cas, vous devez indiquer les paramètres de la grille et fournir ses données,

```
<code>
surface
  grid-units units
  grid origin_x origin_y step_x step_y size_x size_y
  grid-flip [none|vertical|horizontal]
```

```
    data_of_the_grid
endsurface
</code>
```

Dans ce cas, l'origine est le coin inférieur gauche de la grille. Le pas ("step") est la dimension des cellules de la grille, et la taille ("size") et le nombre de cellules dans la ligne (x) et dans la colonne (y). La grille de donnée contient les altitudes (au dessus du niveau de la mer, en mètres). Elles doivent être écrites en lignes (de l'Ouest vers l'Est), en parcourant les lignes du bas (le Sud) vers le haut (le Nord).

*import* (commande du fichier .th)

*import*. Cette commande est utilisée pour importer des données topographiques à partir d'un format externe, "3d", "plt" ou "xyz".

*grade* (fichier de commande .th)

*grade*. Cette commande définit le grade, c.à.d. la précision de la topographie. Se reporter à l'option "sd" de la commande "centerline" (définition de la déviation standard des instruments. Cette commande semble incompatible (d'après la doc) avec la commande sd.

*Syntax : grade <id> ... [<quantity list> <value> <units>] ... endgrade*

*revise* (fichier de commande .th)

*revise*. Cette commande est utilisée pour redéfinir les propriétés d'un objet. Elle peut être sur une seule ligne, ex : "revise id -option1 value1 -option2 value2 ..." ou multilignes, terminées par la commande *endrevise*.

## 2.10 "Therion"

---

*Therion* n'est pas un program graphique, et il peut être exécuté à partir du shell (ligne de commande). La syntaxe est

```
Therion [options] configuration_file
```

Par défaut, le fichier de configuration est *thconfig*. Par exemple *XTherion* invoque *Therion* avec la commande "therion -x *thconfig*".

*Therion* a plusieurs options (qui peuvent être écrites en format long également) :

- -v: affiche la version du programme;
- -h: (help) écrit un court résumé des options disponibles;
- -d: active le mode debug;
- -g: génère un nouveau fichier de configuration;
- -u: met à jour le fichier de configuration;
- -l log\_file: écrit les traces dans le fichier *log\_file*;
- -L: ne génère aucunes traces;
- -q: (silencieux) s'exécute avec un minimum de messages;
- -i: ignore les commentaires lors de l'écriture du fichier de configuration;



- -x: génère le fichier *.xth-thconfig* avec les informations pour *XTherion*
- -p path: indique un répertoire de recherche
- -s file: indique un fichier source.

Il y a aussi des options pour afficher la configuration:

- -print-encodings
- -print-tex-encoding
- -print-init-file
- -print-environment

## 2.11 Syntaxe des fichiers *Survex*

---

Cette section résume la syntaxe des fichiers *Survex*. Reportez vous à [1](#) pour plus de détails.

Les fichiers de données *Survex* sont des fichiers texte avec des lignes de commandes et des lignes de données. Une commande débute par un mot clé, un mot qui commence par une étoile “\*”. La ligne de données contient les données topographiques. Les commentaires sont supportés. Un commentaire débute par un point-virgule, ‘;’, et va jusqu’à la fin de la ligne. Les données qui suivraient sur la ligne sont ignorées.

Le numéro des stations sont référencées dans une colonne qui est la première colonne de gauche en générale, et la colonne de droite est le nom de la station. Par exemple, *lost\_cave.main\_passage.14* est le nom complet de la station “14” dans le fichier topo “*main\_passage*” de la grotte *lost\_cave*”.

Un fichier topographique débute avec la commande “\*begin &lt;survey\_name &gt;” et se termine avec la commande “\*end &lt;survey\_name &gt;”. Les deux commandes prennent en paramètre le nom de la topo, qui doit être identique.

Voici quelques commandes *Survex*.

- *data*. Indique l'ordre des données topographiques dans les lignes des données topo suivantes. Par exemple, *\*data normal from to length clino compass* indique que les données ont un style *normal*, et que vous avez inséré les données topo dans l'ordre suivant: *from\_station*, *to\_station*, distance, pendage, et azimut. Une ligne de donnée suivant cette commande devrait ressembler à ceci *1 2 4.30 17 217*: c'est la visée 1-2 de 4.3 mètres, 17 deg d'inclinaison, et 217 d'azimut
- *entrance* indique qu'un point topo est une entrée. Par exemple *\*entrance 1* indique que le point topo “1” est l'entrée de la grotte.
- *equate* indique l'équivalence de deux stations (par exemple entre deux séances topos). Par exemple *\*equate 1 main\_passage.14* indique que la station “14” du fichier topo “*main\_passages*” coïncide avec la station “1” de la topo courante.
- *include* est utilisé pour inclure un autre fichier de données *Survex*. Par exemple “*\*include main\_passage.svx*” inclus le fichier de donnée *Survex* *main\_passage.svx*.
- *export* liste les stations référencables par d'autres fichiers *Survex*. Par exemple, survey “*main\_passage*” devrait avoir la commande “*\*export 14*” pour rendre la station “14” référencable dans une commande “*\*equate*” d'un autre fichier topo. Ou le fichier global de données peut avoir la commande “*\*export main\_passage.14*” avant (ou après ?) l'*include* du fichier de donnée topo “*main\_passage*”.
- *fix* indique les coordonnées géographiques d'une station. Par exemple “*\*fix 1 ...*” TODO

- *units* indique l'unité de mesure utilisé. Par exemple, *\*units length 0.01 meter* indique que les longueurs sont exprimées en centimètres (0.01 m). Un autre exemple: *\*units compass grad* indique que le compas est gradué en grades.
- *team* indique le nom d'un membre de l'équipe topographique, et optionnellement son rôle. Par exemple. *\*\*team xyz compass clino* indique que "xyz" effectuait les mesures au compas et au clino.

Voici également un exemple simple, le premier fichier topographique contient les données d'une cavité d'exemple utilisée dans ce chapitre. Seul les premières lignes de données ont été incluses. Notez la commande " *export* ", qui rend la station "24" accessible pour les autres fichiers topo.

```
*begin gm-01
*export 24
*data normal from to tape compass clino ignoreall
*units length metres
*calibrate tape 0.0
*units compass degrees
*calibrate compass 0.0
*units clino degrees
*calibrate clino 0.0
;end generated presettings

*units length metres
*title "Grazie Mille"
*team "L. Aimar" compass clino
*team "A. Premazzi" notes pictures
*team "S. Vandone" tape
*team "A. Venturini" tape

; Cap 1 = ingresso
*entrance 1

*data normal from to length clino compass
1 2      5.70   -52   190
2 3      6.50   -50   205
3 4      2.80   -90    0
4 5      3.65   -50   193
5 6      4.35   -33   228
...

*end gm-01
```

Voici le fichier survex global qui incluse trois autres fichiers topographiques et assemble les topos avec des commandes " *equate* ":

```
*begin gm

*data normal from to tape compass clino ignoreall
*units length metres
*calibrate tape 0.0
*units compass degrees
*calibrate compass 0.0
*units clino degrees
*calibrate clino 0.0
```

```
;end generated presettings
```

```
*equate gm-01.24 gm-02.1
```

```
*equate gm-02.13 gm.03.1
```

```
*include gm-01/gm-01.svx
```

```
*include gm-02/gm-02.svx
```

```
*include gm-03/gm-03.svx
```

```
*end gm
```

# UN TUTORIEL THERION

*De PocketTopo à Survex via Therion*

par FOOTLEG

## Préambule

(Il s'agit toujours d'un travail en cours : projet du 15 mars 2016. La localisation en ligne de ce tutoriel est le site Web du UK Cave Surveying Group, où vous pourrez trouver une version mise à jour ou une version terminée à l'adresse : <http://cp.cavesurveying.org.uk/index.php/training>).

Ce tutoriel est conçu pour vous faire suivre les étapes de la création de relevés de grottes à l'aide de *Therion*, en commençant par les données initiales enregistrées sous terre et en terminant par un dessin de relevé fini sous forme de PDF montrant un plan, une coupe et des sections transverses. Il ne traite pas du processus de topographie dans la cavité.

Pour suivre ce tutoriel, vous aurez besoin des éléments suivants :

- Quelques données topo de la grotte (mesures et croquis pris sous terre).
- *Therion* (application de dessin de topographie et de gestion d'une cavité avec visualiseur).
- *PocketTopo* (logiciel de carnet topo électronique fonctionnant sur un PDA *Windows Mobile*. Un PDA ou un ordinateur Windows peut être utilisé pour exécuter le programme afin de suivre ce tutoriel).

Le tutoriel est présenté sous la forme d'une série de leçons, chacune d'entre elles s'appuyant sur les leçons précédentes. Les données du relevé topo réelles utilisées pour présenter les leçons sont disponibles dans le fichier ZIP de téléchargement du didacticiel. Vous préférerez probablement utiliser ces données plutôt que les vôtres, car elles ont été soigneusement choisies pour permettre d'enseigner tous les différents aspects de *Therion* sans avoir à dessiner une quantité énorme de galeries. Si vous commencez à la leçon 1 avec les données brutes et que vous suivez le tutoriel complet, chaque leçon commencera avec les données que vous aurez générées dans la leçon précédente.

Pour vous assurer que vous disposez d'un ensemble de fichiers fonctionnel pour chaque leçon, le fichier ZIP de téléchargement du didacticiel contient l'ensemble des fichiers tels qu'ils devraient être à la fin de chaque leçon. Ainsi, si vous constatez par exemple que vous ne parvenez pas à faire fonctionner quelque chose dans la leçon 5, comparez les fichiers du dossier leçon 5 des données téléchargées avec vos fichiers pour voir quelles sont les différences. (Si vous n'avez jamais utilisé d'outils de comparaison de fichiers, je vous recommande l'outil gratuit *WinMerge*, dont vous trouverez les liens de téléchargement à la fin de ce document). Si vous voulez passer directement à une leçon ultérieure du tutoriel (peut-être avez-vous déjà une certaine connaissance des bases de *Therion*), vous devez commencer la leçon avec les fichiers des données téléchargées pour la leçon numérotée précédente. Par exemple, si vous passez directement à la leçon 8, commencez avec les fichiers du dossier de la leçon 7.

Les fichiers de données de toutes les leçons (et un PDF de ce tutoriel) peuvent être téléchargés à l'adresse suivante : <http://wscc.darkgem.com/footleg/therion/>.

Ce guide de cours a été rédigé en utilisant les versions logicielles suivantes :

- *Therion* (v5.3.16, publié le 30 déc. 2014) <http://therion.speleo.sk>.
- *PocketTopo* (v1.372, publié le 22 août 2014) <http://paperless.bheeb.ch> Application de saisie de données topo et de croquis pour les assistants personnels Windows Mobile.

- *Survex* (v1.2.16, publié le 17 oct. 2014) <http://survex.com> Application de traitement des données de cave et visionneuse.

Pour tirer le meilleur parti de ce tutoriel, vous pouvez également installer les applications suivantes (voir la fin de ce document pour les liens de téléchargement) :

- *Sumatra PDF* (cette visionneuse PDF ne verrouille pas votre fichier PDF lorsque vous le consultez. Vous n'avez donc pas besoin de fermer sans cesse le visualiseur PDF à chaque fois que vous voulez compiler votre projet *Therion*).
- *WinMerge* (Outil de comparaison de fichiers, également connu sous le nom d'outil DIFF. Vous permet de comparer deux fichiers texte et de voir quelles sont les différences. Pratique lorsque vos fichiers de données ne fonctionnent pas comme prévu, pour comparer avec les fichiers de leçons des données téléchargées).
- *TerrainTool* (Crée un maillage de paysage pour montrer la surface au-dessus de votre topographie).
- *GMapCatcher* (Vous permet de télécharger des tuiles cartographiques pour créer une superposition de photos pour votre maillage de surface).

## Plan des leçons de la formation

Les deux premières leçons de ce tutoriel traitent de l'introduction dans *Therion* des données topo de grottes enregistrées avec *PocketTopo* sur un PDA. Le dessin commence dans la leçon 3. Si vous suivez ces leçons en utilisant les données de l'exemple, tout devrait ressembler aux captures d'écran utilisées pour illustrer les leçons. Si vous choisissez d'utiliser vos propres données, vous devrez faire votre propre interprétation de ce à quoi ressemble votre dessin par rapport aux images d'exemple présentées dans ce tutoriel. Si vous travaillez avec des notes sur papier provenant d'une séance topo réalisée avec des notes et des croquis sur papier et au crayon, vous devriez commencer par la leçon alternative 3b pour vous préparer aux leçons de dessin qui suivent.

Les données utilisées pour cette série de leçons proviennent de mon projet de ré-exploration du célèbre Swildon's Hole dans le Somerset, au Royaume-Uni. J'ai choisi cet ensemble de données parce qu'il représente des données du monde réel et certains des problèmes rencontrés lors de l'élaboration d'une topo plus grande à partir d'une séance topo unique et de la combinaison des résultats de plusieurs séances. Il s'agit également d'une grotte avec une forte pente, ce qui permet d'illustrer certains des défis posés par le chevauchement des galeries et la façon dont les fragments (calques) peuvent être organisés pour illustrer proprement la profondeur de la cavité à l'aide de couleurs différentes.

**Leçon 1** : Exporter les données depuis l'application *PocketTopo*

**Leçon 2** : Création d'un projet *Therion* et génération d'un modèle 3D

**Leçon 3** : Dessiner le contour du plan d'un cheminement

**Leçon 3b** : Création d'un projet *Therion* à partir de croquis papier et de notes issues d'un relevé topo

**Leçon 4** : Dessiner des symboles en utilisant d'autres types de lignes

**Leçon 5** : Ajouter des symboles et du texte à l'aide de points

**Leçon 6** : Remplissage de zones

**Leçon 7** : Travailler avec plusieurs calques (scraps)

**Leçon 8** : Structures de données de projets plus complexes

**Leçon 9** : Utiliser les mises en page (layouts) pour changer de style

**Leçon 10** : Contrôler les styles des symboles et utiliser les mises en page prédéfinies

**Leçon X** : Ajouter des sections transverses

## Préambule

Ce tutoriel implique de travailler avec un certain nombre de types de fichiers différents, qui sont tous des fichiers texte. Afin de pouvoir distinguer les différents types de fichiers texte, il est très utile d'afficher les extensions de fichiers sur les ordinateurs *Windows*. Trouvez donc l'option permettant d'afficher les extensions dans l'Explorateur *Windows* (sur les anciennes versions de *Windows*, l'option se trouve sous "Options de dossier et de recherche", dans l'onglet "Affichage" de la boîte de dialogue des paramètres, recherchez la case à cocher "Masquer les extensions pour les types de fichiers connus", et assurez-vous qu'elle n'est PAS cochée. Dans l'explorateur *Windows*, vous devriez voir que les fichiers texte ont un .txt à la fin de leur nom de fichier. Nous allons créer d'autres fichiers contenant du texte brut, mais avec d'autres extensions pour que *Therion* puisse comprendre comment les ouvrir.

Vous devez avoir téléchargé les fichiers du tutoriel pour pouvoir suivre ces leçons. Décompressez le fichier zip téléchargé et vous devriez avoir une série de dossiers de leçons numérotés (chacun d'entre eux contient la structure de fichier que vous devriez avoir à la fin de la leçon à laquelle il se rapporte). Vous pouvez trouver tous les fichiers de données *PocketTopo* utilisés dans ce tutoriel dans le dossier 'PocketTopo Data'. Nous commencerons la leçon 1 avec l'un de ces fichiers, et à la fin de la leçon, vous devriez avoir un dossier de projet contenant un certain nombre de fichiers et de dossiers identiques à ceux du dossier de la leçon 1 du téléchargement

## Leçon 1: Exporter les données de l'application *PocketTopo*

La première série de leçons de ce tutoriel est basée sur une seule séance topo d'une galerie simple de la cavité. Cela permettra de disposer de données simples pendant que nous apprenons les bases du dessin, avant que les leçons suivantes n'introduisent plus de problèmes provenant de parties plus complexes de la cavité. Les données du relevé et les dessins réalisés sous terre sur le PDA sont tous enregistrés dans un fichier *PocketTopo* avec l'extension '.top'. Nous devons exporter ces données dans un format qui capture toutes les informations dans des fichiers en texte clair qui peuvent être lus sur n'importe quel ordinateur.

Ces fichiers seront utilisés pour importer les données dans *Therion*, mais aussi pour archiver les données dans un format qui sera lisible par des personnes dans le futur sans avoir besoin de *PocketTopo* et d'un ordinateur capable de le faire fonctionner. Vous pouvez compléter cette première leçon soit en utilisant *PocketTopo* sur un PDA *Windows Mobile*, soit en exécutant le programme *PocketTopo* sur un ordinateur *Windows*. Les étapes suivantes supposent que vous utilisez *PocketTopo* sur un PDA. Si vous l'utilisez sur un ordinateur, au lieu de sauvegarder sur une carte mémoire, vous pouvez sauvegarder les fichiers d'exportation directement sur votre disque dur.

Ouvrez le fichier 'ShortDryWay.top' dans *PocketTopo* (en utilisant le menu Fichier/Ouvrir). Vous devriez voir les chiffres représentant les données du relevé, et aussi être en mesure de visualiser les croquis dans PocketTopo.

Sélectionnez Fichier/Exporter/Texte dans le menu. Dans la fenêtre qui s'ouvre, changez l'emplacement pour une de vos cartes mémoire (plutôt que la mémoire principale), choisissez un dossier sur votre carte mémoire et entrez "ShortDryWay" dans le champ du nom. Cliquez ensuite sur Enregistrer. Un fichier nommé "ShortDryWay.txt" sera alors enregistré sur votre carte mémoire. Ce fichier contient toutes les données du relevé topo (les chiffres), y compris la date de la séance, la déclinaison magnétique et tout commentaire ajouté sur l'écran de données.

Sélectionnez File/Export/*Therion* dans le menu. Dans la fenêtre qui s'ouvre, changez l'emplacement pour une de vos cartes mémoire (plutôt que la mémoire principale), choisissez un dossier sur votre carte mémoire et entrez "ShortDryWayth" dans le champ du nom. Cliquez ensuite sur Enregistrer. Un fichier nommé " ShortDryWayth.txt " sera alors enregistré sur votre carte mémoire. Ce fichier contient à la fois les données et les dessins dans un format qui peut être importé dans le logiciel *Therion*. Nous avons ajouté le 'th' à la fin du nom du fichier afin de ne pas écraser le fichier texte précédent.

Nous n'avons pas besoin du premier fichier d'exportation texte pour l'importer dans *Therion*, car le deuxième fichier d'exportation *Therion* contient tout ce dont *Therion* a besoin. Mais les commentaires ne sont édités que dans le fichier d'exportation Texte. Pour conserver ces informations importantes, nous devons donc exporter les deux fichiers. Si vous êtes curieux, vous pouvez ouvrir les deux fichiers .txt en cliquant sur le lien suivant

Nous devrions maintenant avoir toutes nos données sauvegardées sur la carte mémoire du PDA. (Ceci en supposant que vous ayez conservé le fichier .top original sur la carte mémoire pendant le relevé topo. Il est recommandé de le faire car cela vous permet de toujours récupérer vos données si votre PDA a été endommagé sous terre.

Vous pouvez toujours retirer la carte mémoire d'un PDA mort et la lire sur un autre ordinateur. Mais vous ne pourrez pas lire la mémoire intégrée du PDA si l'appareil est cassé).

## **Organisation des dossiers**

Les procédures suivantes sont des recommandations plutôt qu'une méthode de travail imposée. L'important est de préserver vos fichiers de données brutes, car ils représentent le dur labeur du relevé topo réalisé sous terre. Tout le reste peut être récupéré à partir de ces fichiers si vous ne les avez pas perdus ou modifiés.

Nous allons donc les placer dans leur propre dossier pour les garder en sécurité pendant que nous travaillons sur les dessins. Créez un dossier principal pour votre projet. Nous avons appelé le nôtre " Swildons Hole ".



À l'intérieur de ce dossier, créez un deuxième dossier nommé "PocketTopo" et copiez les fichiers de la carte mémoire de votre PDA dans ce dossier PocketTopo sur votre ordinateur. Vous devriez y trouver les fichiers suivants :

- ShortDryWay.top
- ShortDryWay.txt
- ShortDryWayth.txt

Ce sont les fichiers de données originaux et l'objectif est que ceux-ci ne soient jamais édités directement. Nous les sauvegardons pour les garder en sécurité. Nous sommes maintenant prêts à commencer l'apprentissage de *Therion*.

## Leçon 2 : Création d'un projet *Therion* et génération d'un modèle 3D

Dans cette leçon, nous allons créer un nouveau projet *Therion*, importer nos données top et créer un modèle de la ligne centrale de cheminement de la cavité.

Si vous n'utilisez pas *PocketTopo* pour enregistrer vos données topo, vous ne pourrez pas suivre les étapes d'importation de données de cette leçon. Vous devrez alors saisir vos données directement dans l'éditeur *Therion*. Regardez le fichier .th dans le dossier de la leçon 2 pour voir comment les données de la ligne de cheminement sont formatées, ou utilisez le fichier de données *PocketTopo* fourni pour suivre cette leçon en utilisant les données de l'exemple.

Lancez l'application *XTherion*. Il s'agit de l'éditeur *Therion*. Il y a trois fenêtres principales dans l'éditeur. L'éditeur de texte, l'éditeur de carte et le compilateur. Vous pouvez passer de l'une à l'autre en sélectionnant les éléments appropriés dans le menu Fenêtre, en utilisant les boutons de la barre d'outils située en haut de l'application ou en appuyant sur l'une des touches de fonction F1, F2 ou F3.

En commençant par l'éditeur de texte (appuyez sur F1 pour passer à cette vue), nous allons créer le fichier de projet principal pour notre topo. Sélectionnez " Fichier/Nouveau " dans le menu Fichier pour créer un nouveau fichier, puis " Fichier/Enregistrer sous " pour enregistrer le fichier vide. Nommez ce fichier 'swildons.th' et enregistrez-le dans le dossier principal que nous avons créé pour notre projet (le dossier que nous avons nommé 'Swildons Hole').

**Remarque** : tout au long de ce tutoriel, nous utiliserons des noms de fichiers sans espace. Cela facilite l'apprentissage car cela évite de devoir mettre les noms de fichiers entre guillemets lorsqu'on y fait référence dans d'autres fichiers. Lorsque nous devons inclure des fichiers de dessin dans notre projet, ou spécifier le nom d'un fichier de sortie que nous voulons que *Therion* génère, nous pouvons simplement entrer le nom du fichier. Si vous travaillez avec des fichiers dont le nom comporte des espaces, vous devrez mettre le nom du fichier entre guillemets partout où il est inclus dans un autre fichier. (par exemple, "Swildons Hole.3d").

Nous devons maintenant définir notre topographie dans le fichier .th. Entrez le texte suivant dans la fenêtre de l'éditeur :

```
survey
swildons
endsurvey
```

Cela permet de définir une topographie nommée 'swildons' dans laquelle nous devons placer nos données relevées. Nous pouvons l'importer directement depuis le fichier d'exportation *Therion* que nous avons exporté depuis *PocketTopo*. Cliquez entre les lignes " survey " et " endsurvey " pour y placer le curseur, puis sélectionnez " File/Import " dans le menu. Sélectionnez le fichier d'exportation *Therion* que nous avons créé dans la leçon précédente (ShortDryWayth.txt). Lorsque ce fichier est importé, vous devriez voir qu'un bloc de données de ligne centrale a été créé entre les lignes 'survey' et 'endsurvey'. Ce bloc de données commence par 'centerline' et se termine par 'endcenterline'. Enregistrez le fichier (Fichier/Enregistrer).

Nous avons maintenant dans notre fichier tout ce dont nous avons besoin pour construire notre premier modèle de la cavité. Pour ce faire, nous devons créer un fichier dans la fenêtre du compilateur pour décrire le type de sortie que nous voulons créer.

### Compiler le projet *Therion*

Passez à la fenêtre du compilateur (appuyez sur F3 ou utilisez le bouton ou les menus de la barre d'outils pour changer de fenêtre). Nous devons créer un fichier de configuration pour indiquer à *Therion* ce que nous voulons qu'il fasse avec les données de notre relevé topo. Créez un nouveau fichier en utilisant le menu 'File/New'. Le nom de fichier par défaut est '*thconfig*'. Changez-le en '*thconfig.thc*' et enregistrez-le dans le dossier du projet (que nous avons appelé 'Swildons Hole').

Vous devriez maintenant pouvoir cliquer dans le volet supérieur de la fenêtre du compilateur et saisir dans cette fenêtre. Entrez le texte suivant :

```
source swildons.th
export model -fmt survex -o swildonsth.3d
```

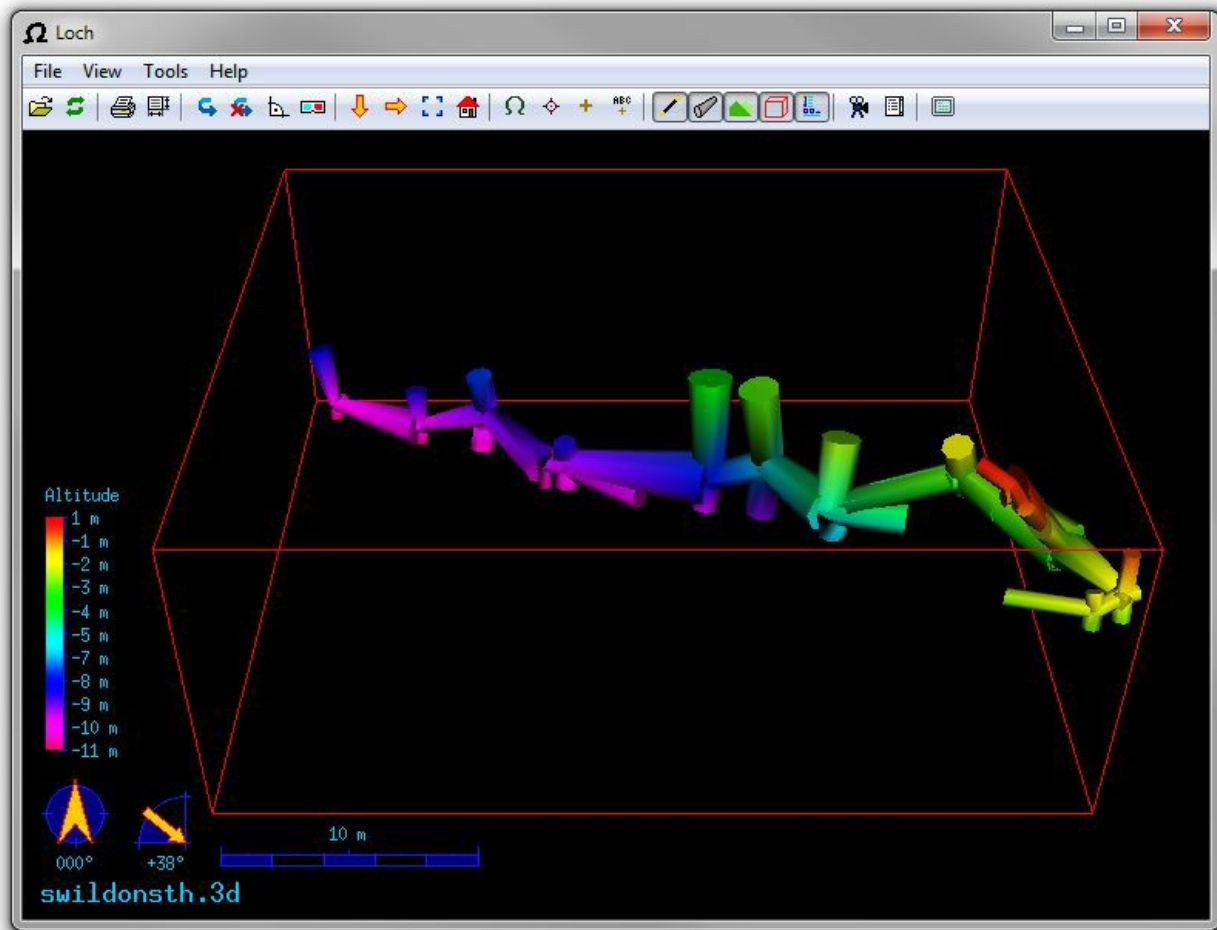
Cette paire d'instructions indique d'abord à *Therion* le nom du fichier de notre projet à traiter ('swildons.th'), puis lui demande de générer un modèle de la cavité. Nous demandons que le format du modèle soit un fichier *Survex 3D*, et nous donnons le nom du fichier de sortie 'swildonsth.3d'.

Ce nom de fichier peut être quelconque, mais ici nous décidons arbitrairement de le nommer de la même manière que notre projet 'swildons' plus un 'th' pour indiquer que ce fichier a été généré à partir de *Therion* et non directement à partir de *Survex*.

Nous avons maintenant fourni toutes les informations dont *Therion* a besoin pour générer une sortie. Cliquez sur le bouton de compilation (qui ressemble à une roue dentée noire dans la barre d'outils) ou appuyez sur F9. Si tout est en ordre, vous devriez voir une barre verte OK dans le panneau de droite, et une quantité de texte de sortie dans le panneau inférieur de la fenêtre de compilation. Vous devriez également constater qu'un nouveau fichier nommé 'swildonsth.3d' est apparu dans le répertoire de votre projet.

Double-cliquez pour ouvrir ce fichier dans *Aven* (le visualisateur de grottes de *Survex*) et vous pouvez faire pivoter et zoomer sur votre modèle pour voir votre cavité. Vous pouvez également lancer la visionneuse *Therion 'Loch'* à partir du groupe *Therion* de votre menu de démarrage et ouvrir le fichier .3d dans celle-ci.

Le visualiseur *Loch* dispose d'une fonction permettant d'extrapoler les parois des galeries pour les projets de topographie qui ne contiennent pas ces informations. Notre fichier 3D ne contient pas de données sur les dimensions des galeries à ce stade, mais nous pouvons demander à *Loch* d'extrapoler ces informations à partir des données de ligne centrale de cheminement afin d'obtenir un modèle 3D solide. Dans le visualiseur, sélectionnez 'Outils/Options' dans les menus et choisissez l'option 'Extrapoler uniquement les fichiers sans informations sur les parois'. Vous ne verrez pas l'effet jusqu'à ce que vous rechargez le modèle (il y a un bouton de rechargement sur la barre d'outils, ou vous pouvez le faire via 'File/Open' ou 'File/Reload' dans les menus. Nous devrions maintenant voir un modèle 3D solide. Ce n'est qu'une approximation de la largeur et de la hauteur de la galerie de la grotte à ce stade, mais cela s'améliorera au fur et à mesure que nous ajouterons des informations au projet *Therion*. Vous devriez voir quelque chose comme ceci dans *Loch*.



### Ajout d'informations supplémentaires aux données du relevé topo

Les données numériques essentielles pour la ligne de cheminement de la cavité ont été automatiquement importées de *PocketTopo*, mais il est généralement bon d'inclure des données supplémentaires pour enregistrer l'identité des topographes. Il est également utile d'étiqueter les entrées des cavités car elles

peuvent être mises en évidence sur les modèles. Si vous revenez à la vue de l'éditeur de texte (F1) et que vous regardez le bloc de données de la ligne centrale, vous verrez que la date de la séance a déjà été importée. Sous la date se trouve une ligne de données détaillant l'ordre dans lequel les données numériques sont fournies. Nous allons insérer quelques informations supplémentaires entre ces deux lignes.

Cliquez donc pour placer le curseur à la fin de la ligne de la date et appuyez sur la touche 'Enter' plusieurs fois pour créer deux lignes vides après la date. Puis saisissez les lignes suivantes :

```
team "Joe Bloggs"  
instruments team "Jane Bloggs"  
notes team "Tim Bloggs" dog
```

Ces lignes fournissent des informations sur les topographes. La personne qui lit les instruments, la personne qui enregistre/dessine le relevé sur le PDA (notes) et la personne qui a couru partout pour installer les stations, etc. (le chien). Pour les relevés topographiques traditionnels au décamètre à ruban, vous pouvez également indiquer un membre de l'équipe suivi du mot "ruban". Il existe d'autres rôles et de nombreuses autres informations que vous pouvez fournir (quels types d'instruments ont été utilisés, auteur, informations sur le copyright, etc.) Consultez le [Therion Book](#) pour obtenir des détails sur tous les rôles supportés et les autres informations qui peuvent être ajoutées dans un bloc de données de ligne centrale de cheminement (cherchez 'centerline' dans la table des matières). Notez que les noms des membres de l'équipe ne peuvent contenir qu'un seul caractère d'espacement, utilisez donc des tirets pour les noms composés ou votre projet générera une erreur lorsque vous essaierez de le compiler.

Nous voulons également indiquer quelles stations de relevé se trouvent à l'entrée de la cavité. Dans notre cas, aucune de nos stations ne se trouvait à une entrée réelle, mais pour les besoins de la démonstration, nous allons prétendre que la station 3.0 était à l'entrée de la grotte. Nous pouvons l'indiquer avec la ligne suivante :

```
station 3.0 "main ent." entrance
```

Notez que nous devons fournir un commentaire, sinon cette ligne ne fonctionnera pas. Nous avons mis le commentaire 'main ent..' entre guillemets. Si vous ne voulez pas de commentaire, vous aurez toujours besoin des guillemets, mais vous pouvez les saisir sans texte entre eux.

```
station 3.0 "" entrance
```

Recompilez maintenant le projet (F9 ou cliquez sur le bouton noir de la roue dentée de la barre d'outils). Ouvrez à nouveau le modèle dans le visualiseur et vous pouvez maintenant activer l'option de visualisation des entrées et vous verrez l'entrée en surbrillance dans le visualiseur d'*Aven*. (Mais pas dans la visionneuse *Loch*, car elle ne détecte pas le marqueur d'entrée des fichiers modèles *Survex*. Nous réglerons ce problème sous peu lorsque nous apprendrons comment sortir des modèles directement pour *Loch*).

Il est également très utile de donner l'emplacement de la grotte, car elle peut alors être positionnée par rapport à d'autres endroits dans le monde. Cela est nécessaire si vous voulez positionner la grotte sur *Google Earth*, ou sous une superposition de paysage de surface, ou dans le même modèle que d'autres grottes proches. Nous pouvons spécifier la position d'une station en utilisant la commande `fix`. Nous traiterons des systèmes de coordonnées plus tard, lorsque nous construirons un modèle KML à visualiser dans *Google Earth*. Mais pour l'instant, nous allons simplement donner les coordonnées de la position de l'une de nos stations. La station d'entrée est la plus logique car nous pouvons la positionner à l'aide d'un GPS. Jusqu'à ce que nous spécifions le contraire, les unités de toutes les coordonnées que nous utilisons sont supposées être des mètres. Nous allons donc entrer la position en utilisant la référence OS Grid dans le carré de 10 km de la grille OS où se trouve notre grotte, en nous basant sur la position GPS et l'altitude de la station d'entrée. L'ordre doit être Easting, Northing, Elevation. L'entrée de notre grotte se trouve à l'emplacement de la grille OS ST 53122 51308, à environ 237 m au-dessus du niveau de la mer. Pour spécifier cette position en mètres, nous entrons la ligne suivante :

```
fix 3.0 53122 51308 237
```

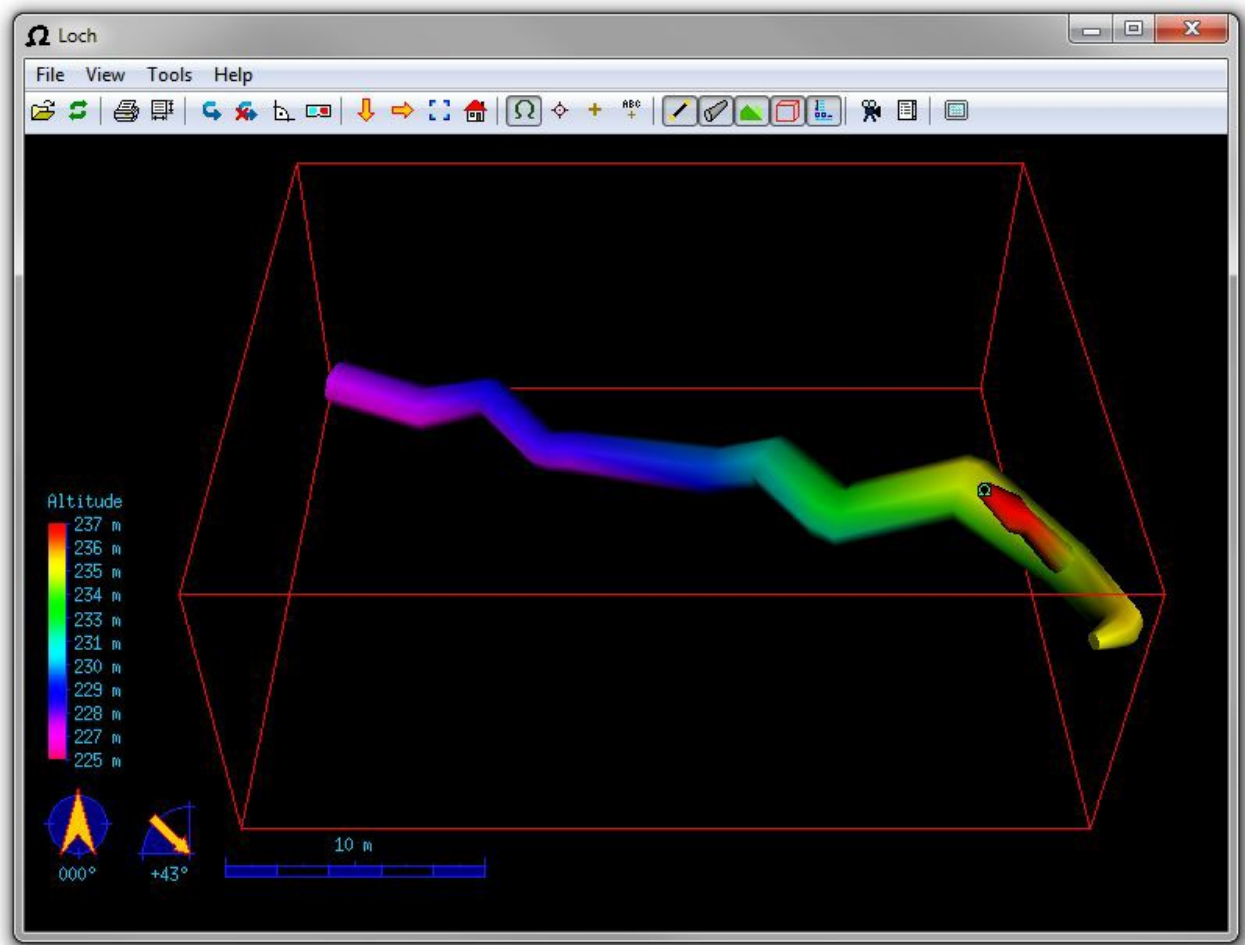
Maintenant, si nous compilons notre projet, notre modèle *Survex* nous donnera les coordonnées de grille et l'altitude correctes pour n'importe quelle station lorsque nous le visualiserons dans le visualiseur *Aven*. Plus important encore, si nous incluons une deuxième grotte dans le même modèle, les deux grottes seront positionnées correctement l'une par rapport à l'autre, ce qui nous permettra de voir à quelle distance se trouvent les parties les plus proches de chaque grotte.

*Therion* peut produire une grande variété de modèles différents. Nous allons en générer d'autres pour compléter cette leçon. Le modèle par défaut de *Therion* est le modèle *lox*. C'est le modèle pour lequel le *Loch viewer* est principalement conçu. Nous devons donc à nouveau passer à la fenêtre du compilateur (F3) et entrer la ligne suivante pour demander au compilateur de construire un modèle *lox* :

```
export model -o swildons.lox
```

Maintenant, si nous compilons à nouveau le projet, nous devrions trouver un nouveau fichier de sortie 'swildons.lox'.

Double-cliquez sur ce fichier dans l'Explorateur *Windows* pour l'ouvrir dans la visionneuse *Loch*. Le modèle semble un peu mieux maintenant que nous avons généré le modèle par défaut pour *Therion*. Notez également que l'entrée peut être affichée (cliquez sur le bouton 'Show entrances' de la barre d'outils), et que la gamme d'altitude reflète maintenant l'altitude réelle des stations parce que nous avons fixé une station à l'altitude de l'entrée.



Tout ce que *Therion* sait à ce stade, c'est que nos données de relevé topo sont situées sur une grille d'un mètre à un ensemble de coordonnées spécifié. Nous devons indiquer à *Therion* le système de quadrillage que nous utilisons pour qu'il puisse ajuster la référence au Nord du Nord magnétique (c'est-à-dire le Nord mesuré par notre boussole pointant vers zéro degré) au Nord du quadrillage que nous utilisons (c'est-à-dire le Nord tel que représenté par le quadrillage sur une carte). Cela permettra également à *Therion* de localiser le modèle à l'endroit correct sur la planète. Nous spécifions le système de coordonnées de la grille avec la ligne suivante :

```
cs OSGB:ST
```

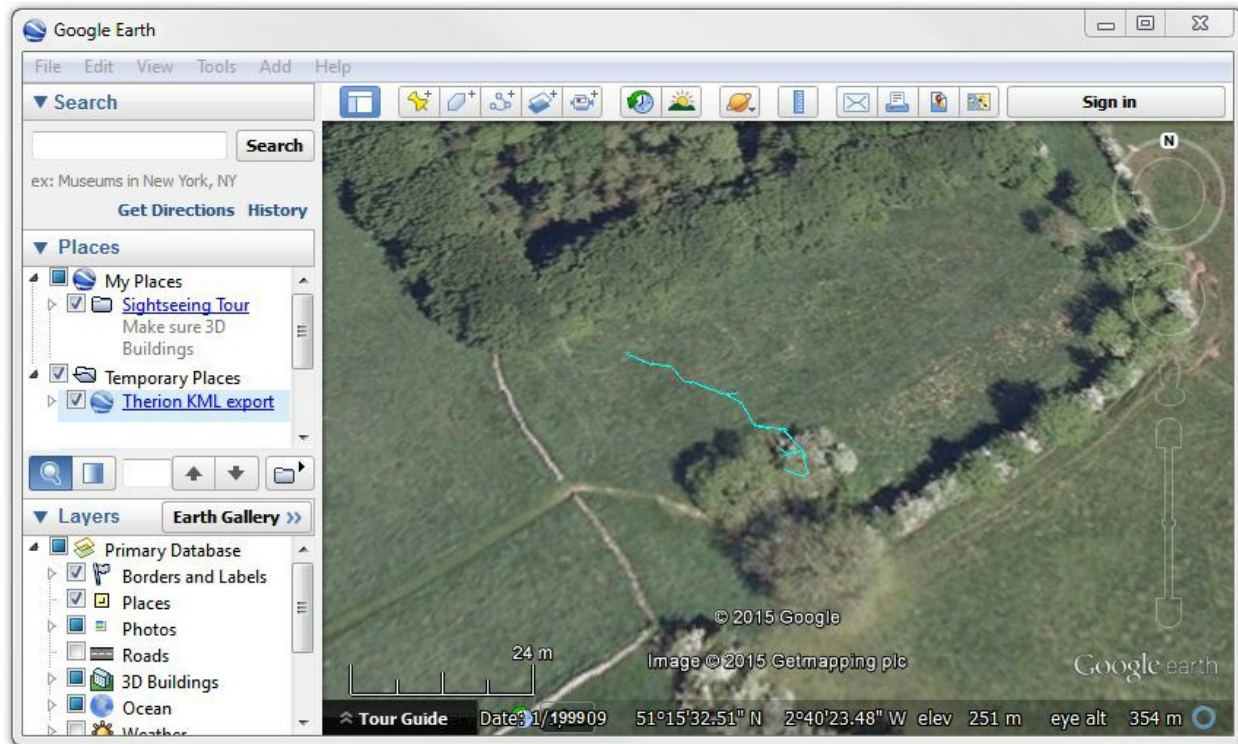
J'ajoute généralement cette ligne immédiatement après la déclaration d'ouverture '[centerline](#)', car il s'agit d'une information importante concernant les données de la topo. Mais tant qu'elle se trouve entre les lignes '[centerline](#)' et '[endcenterline](#)' dans votre fichier de données, *Therion* devrait la traiter correctement.

Maintenant nous pouvons générer un modèle de ligne centrale pour *Google Earth*. Ajoutez la ligne suivante dans la fenêtre du compilateur :

```
export model -fmt kml -o swildons.kml
```

Maintenant, si nous compilons à nouveau le projet, nous devrions trouver un nouveau fichier de sortie 'swildons.kml'. Double-cliquez sur ce fichier dans l'explorateur *Windows* pour l'ouvrir dans *Google Earth* (en supposant que vous avez installé *Google Earth* que vous pouvez télécharger sur Internet). Vous devriez voir votre modèle de cheminement de la cavité situé à l'entrée réelle de Swildons Hole. (Nos données de relevé topo ne concernent pas réellement les vraies galeries d'entrée car elles sont trop complexes pour une leçon de démarrage, mais la station que nous avons marquée comme "entrée" dans notre projet devrait se trouver dans les arbres où se trouve l'entrée réelle de la grotte).

(Capture d'écran de *GoogleEarth*. Le Copyright de l'image est visible)



Les fichiers du projet final pour cette leçon ont été copiés dans le dossier Leçon 2 du fichier zip de données de projet fourni. Si vous avez des doutes ou si vous trouvez que votre projet ne fonctionne pas, utilisez ces fichiers comme référence pour savoir à quoi doit ressembler le projet complet à ce stade. Dans la prochaine leçon, nous commencerons à dessiner le croquis du plan de notre relevé topographique.

### Leçon 3: Dessiner le contour du plan d'un cheminement

Dans cette leçon, nous allons créer un nouveau fichier d'esquisse dans notre projet *Therion* en important notre plan topographique de PocketTopo dans l'éditeur *Therion*, et en le reliant à la ligne centrale de cheminement de notre cavité.

Avec les fichiers de projet créés dans la leçon précédente ouverts dans l'application *XTherion*, nous allons maintenant ajouter notre premier fichier de dessin dans le projet. Ouvrez l'éditeur de carte (appuyez sur F2

pour passer à cette vue), et créez un nouveau fichier en utilisant le menu 'Fichier/Nouveau'. Enregistrez ce fichier dans le dossier du projet (que nous avons appelé "Swildons Hole"). Le type de fichier pour les esquisses est .th2, nous allons donc nommer ce fichier 'swildons.th2'.

*Dans l'étape suivante, nous allons importer le croquis du plan réalisé dans PocketTopo. Si vous utilisez un croquis réalisé sur papier plutôt que sur un PDA, vous devriez passer à la leçon 3b à ce stade, pour apprendre comment importer des croquis scannés pour les dessiner.*

*Revenez ensuite à cette leçon et reprenez après le paragraphe suivant où nous introduisons le concept de "scraps". (La leçon 3b n'a pas encore été écrite dans ce projet tutorial).*

Dans le menu "Editer" de l'éditeur de carte, sélectionnez "Insérer une image" (tout en bas, pas dans le sous-menu "Insérer"). Dans la fenêtre de dialogue qui s'affiche, changez les "types de fichiers à afficher" de "Images" à "Exportation PocketTopo *Therion*" dans la liste de sélection située en bas à droite de la boîte de dialogue. Maintenant, naviguez vers le dossier où nous avons placé nos fichiers exportés de *PocketTopo*, sélectionnez le fichier 'ShortDryWayth.txt' et cliquez sur 'Ouvrir'. Il vous sera demandé de confirmer certaines propriétés de 'XVI'. Assurez-vous que 'Plan' est sélectionné car nous voulons importer le croquis du plan. Laissez l'échelle (1:200), la résolution (200dpi) et l'espacement de la grille (1.0m) aux valeurs par défaut indiquées ici entre parenthèses et cliquez sur 'OK'. Vous devriez maintenant voir le dessin du plan que vous avez fait dans PocketTopo affiché dans l'éditeur de carte.

Les dessins dans *Therion* sont constitués de parties appelées scraps. Pour des grottes très simples, il peut être possible de dessiner l'ensemble du relevé en plan en un seul morceau ou claqué, mais dans la plupart des cas, plusieurs morceaux seront nécessaires. Il y a quelques règles de base à suivre pour décider de la façon de diviser un relevé en morceaux.

1) Aucun fragment ne peut se chevaucher. Ainsi, si votre grotte contient une galerie qui se trouve au-dessus ou au-dessous d'une autre galerie, vous devez dessiner chacun des passages qui se chevauchent dans leur propre scrap.

2) Les fragments ne peuvent pas être trop grands. Qu'entend-on par "trop grand" ? Vous le saurez si vous essayez de dessiner trop de galeries sur un même scrap, car *Therion* générera une erreur lorsque vous tenterez de créer le relevé. Si cela se produit, vous pouvez toujours diviser le dessin en plusieurs morceaux à ce moment-là. Il n'y a donc pas lieu de s'inquiéter de ce problème maintenant. Notre travail pour ce tutorial ne sera pas aussi grand.

3) Pour les fonctions telles que la couleur en fonction de l'altitude, l'ensemble du scrap sera ombré de la même couleur, c'est donc une bonne idée de commencer un nouveau scrap s'il y a un changement significatif de hauteur dans la partie de la grotte que vous dessinez. Une bonne règle consiste à commencer un nouveau scrap si la hauteur a changé de plus de 5 mètres d'une partie de votre scrap à l'autre. Mais utilisez votre jugement pour déterminer ce qui convient le mieux à votre cas particulier. Essayez de joindre les morceaux de calque (scrap) à des endroits où il n'y a pas de détails de galerie autres que les parois, car *Therion* fera alors un meilleur travail pour créer une jonction transparente entre eux.



Nous commençons par créer un nouveau scrap pour dessiner les parois de la grotte de la salle principale de notre croquis. En bas à droite de la fenêtre de l'éditeur de carte se trouve un certain nombre d'onglets avec des barres bleues qui les séparent. Faites défiler l'écran jusqu'en haut et cliquez sur l'intitulé de l'onglet "Objets" pour l'ouvrir. À ce stade, il ne devrait contenir qu'une seule ligne de texte indiquant "end of file". Cette partie affiche tous les objets de notre dessin, y compris les scraps. Notre première tâche consiste donc à créer un nouveau morceau de calque (scrap). Vous pouvez le faire en cliquant sur l'icône "Insert new scrap" dans la barre d'outils au-dessus de ce panneau, ou en utilisant le raccourci clavier "Ctrl+R".

#### Insert new scrap button

Dans les panneaux de droite, le focus doit passer au panneau "Scraps" où vous pouvez définir les options d'un scrap. Nous devons lui donner un nom d'identification dans le champ "id". Par défaut, il sera nommé comme "scrap1". Il sera très utile plus tard d'utiliser un système pour nommer les calques (et les autres objets de nos dessins). Nous dessinons un plan dans ce calque, donc commencez par le nom de la grotte (ou une partie de la grotte pour les topos plus importantes). Terminez le nom par 'S' (pour scrap) suivi de 'P' (pour plan), et terminez par un numéro pour ce scrap. Ainsi, nous appellerons notre premier plan scrap pour Swildons Hole 'swilSP1'. Enfin, nous devons indiquer à *Therion* de quel type de scrap il s'agit. Sélectionnez donc 'plan' dans la case 'projection' située sous le champ 'id'.

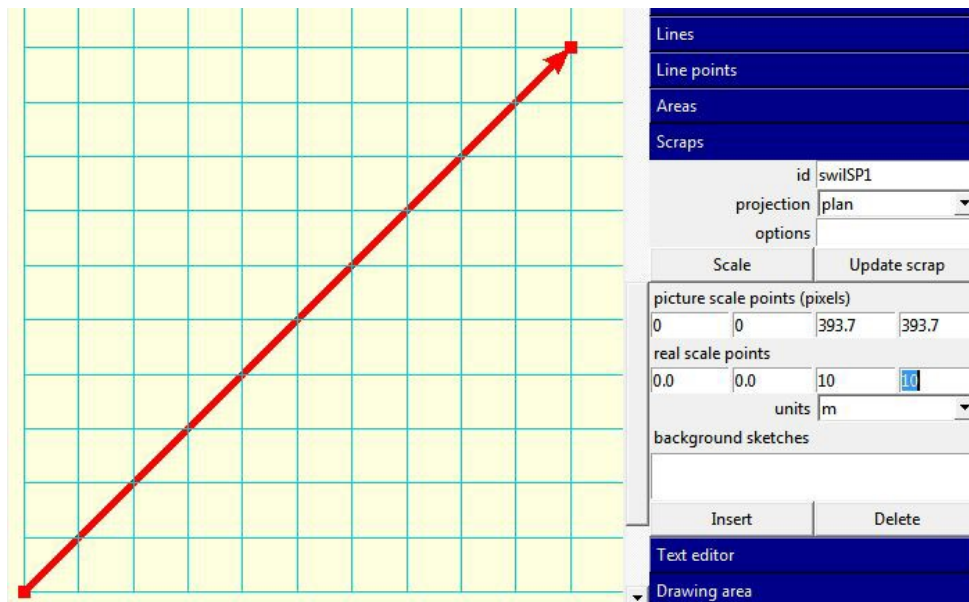
Avant de commencer à dessiner, il est utile d'indiquer à *Therion* l'échelle de notre croquis. *Therion* peut calculer cette échelle pour les scrap qui contiennent plus d'une station de relevé (parce que *Therion* connaît la distance entre les stations de relevé et les données de relevé). Mais quand il s'agit de dessiner des coupes transverses aux stations, il n'y aura qu'une seule station dans chaque scrap. Il est donc préférable de définir l'échelle maintenant, car elle sera copiée sur les autres scraps du fichier de dessin lorsque nous les ajouterons plus tard. Vous verrez les champs d'échelle dans le panneau d'information sur le scrap où nous définissons l'identifiant et le type de scrap. Les champs d'échelle nécessitent l'identification d'une paire de points sur le dessin, et l'échelle réelle de ces points en dessous. Vous verrez qu'il y a un bouton avec le texte "Échelle" sur le panneau d'information des scraps. Si vous cliquez dessus, une barre rouge apparaît en bas de la fenêtre, indiquant que nous sommes en mode "Scraps". Nous pouvons maintenant cliquer à deux endroits sur le dessin pour marquer le début et la fin de la flèche de mise à l'échelle. En utilisant la grille affichée en bleu à l'écran, cliquez sur le coin inférieur gauche de la grille, puis à nouveau sur le point qui se trouve à 10 cases en largeur et à 10 cases en hauteur du coin inférieur gauche. Vous devriez maintenant voir une flèche rouge entre ces deux points.

Les champs de points d'échelle de l'image devraient maintenant afficher des nombres allant approximativement de 0,0,0,0 à 395,395. Étant donné que nous avons importé notre dessin de *PocketTopo*, nous pouvons déterminer les valeurs exactes que doivent prendre ces champs. Définissez donc les valeurs 0,0 et 393,7,393,7 et cliquez sur le bouton "Update scrap". La flèche rouge doit maintenant être parfaitement alignée avec l'origine de la grille bleue et le point situé à 10 m en face et à 10 m au-dessus.

Nous devons également définir les valeurs des points d'échelle réelle pour indiquer les coordonnées de ces points dans le monde réel, c'est-à-dire 0,0 10,10 avec des unités définies sur 'm'.

Si vous utilisez un dessin papier scanné plutôt qu'un croquis importé par PocketTopo, consultez la leçon 3b pour savoir comment configurer l'échelle de scrap pour votre dessin. Les chiffres spécifiques utilisés ici ne s'appliquent qu'aux dessins PocketTopo, qui ont tous la même échelle.

Notre boîte d'information sur les calques devrait ressembler à ceci :

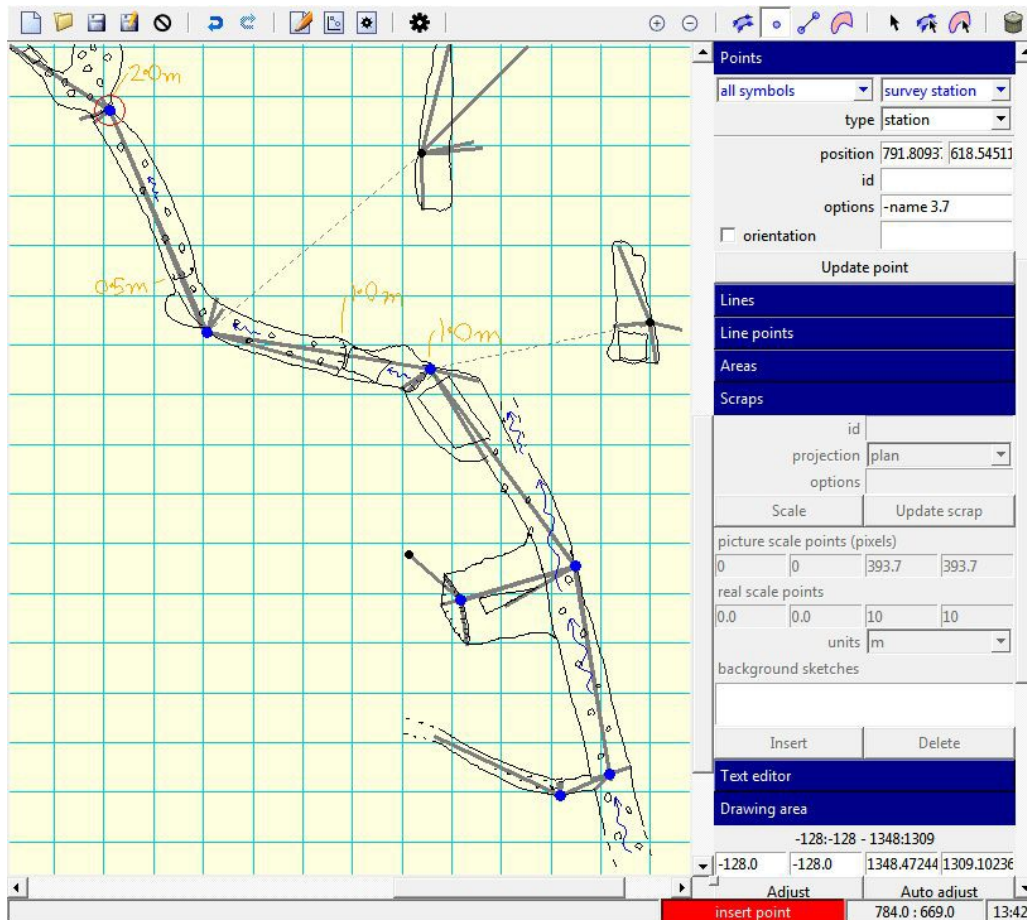


Ensuite, nous devons marquer les stations topo sur notre dessin. L'éditeur de cartes *Therion* nous donne ici un coup de main car notre dessin a été importé de *PocketTopo*. Passez en mode "insertion de point" en cliquant sur l'icône du point bleu dans la barre d'outils.



La barre d'état du mode située au bas de l'écran doit devenir rouge et afficher "insert point". Lorsque la barre d'état du mode est rouge, cela indique que nous sommes en mode dessin et que tout clic sur la zone de dessin entraînera la création d'objets dans le dessin. Nos stations d'arpentage sont déjà indiquées sur le dessin de PocketTopo par des points noirs. Cliquez sur l'un d'entre eux. Il devrait devenir bleu et dans la boîte d'information sur les points à droite, vous devriez voir qu'un point de station de relevé a été créé automatiquement et qu'on lui a donné le nom de la station de notre relevé PocketTopo. Nous sommes toujours en mode insertion de points, alors cliquez sur les stations restantes le long des principaux segments de l'étude le long de la chambre. N'incluez pas les stations sur les coupes transverses à ce stade (nous les ajouterons dans différentes coupes transverses dans une leçon ultérieure). Lorsque vous passez la souris sur chaque station, la barre d'information en bas à gauche de l'éditeur affichera le numéro de la station pour ce point. Les stations que nous voulons inclure dans ce premier scrap sont 3.1 à 3.13. L'ordre dans lequel vous cliquez sur les stations n'a pas d'importance car les croquis importés de PocketTopo incluent les numéros des stations dans le dessin, et l'éditeur saura donc quel numéro donner à chaque station. N'oubliez pas que si vous faites des erreurs, vous pouvez revenir en arrière en sélectionnant

"Edit/Undo" dans le menu, ou en appuyant sur Ctrl+Z pour annuler. Une fois l'insertion des points terminée, quittez le mode d'insertion en appuyant sur la touche "Esc". La barre d'état du mode doit redevenir verte.



Insertion de points des stations topo

Nous pouvons maintenant commencer à dessiner les parois. Nous utilisons pour cela des lignes. Dans *Therion*, une ligne de paroi a un intérieur et un extérieur, indiqués par une petite marque jaune à l'extrémité de la ligne. Les coches doivent toutes pointer vers l'"air libre" à l'intérieur de la galerie de la cavité, et non vers la roche en place. Si une ligne est tracée dans le mauvais sens, nous pouvons l'inverser pour placer la coche du bon côté. Mais nous pouvons éviter d'avoir à le faire pour toutes nos lignes en dessinant autour des parois de la galerie dans le sens inverse des aiguilles d'une montre. Passez en mode "insertion de ligne" en cliquant sur le bouton "Insérer une nouvelle ligne" de la barre d'outils.

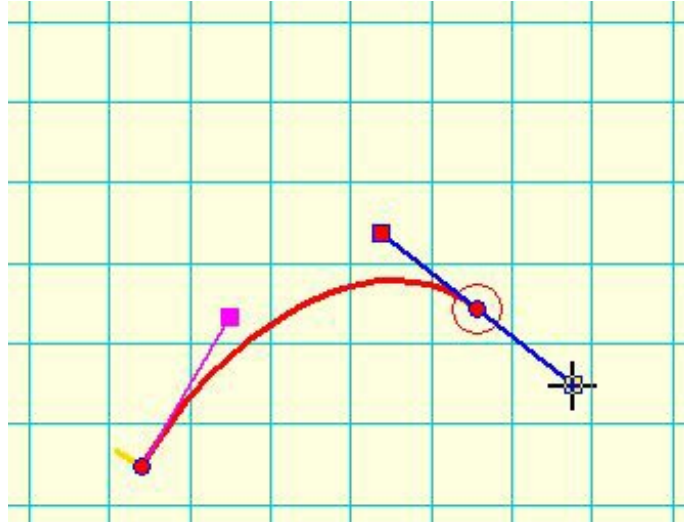


La barre d'état du mode devrait devenir rouge et afficher le message "insérer un point de ligne".

Dessiner des lignes dans l'éditeur de cartes de *Therion* demande un peu d'entraînement. Chaque ligne est constituée d'une série de points, et chaque point est associé à une paire de points de contrôle pour contrôler la courbe de la ligne entre les points. Ce type de ligne incurvée avec des points de contrôle est connu sous le nom de courbe de Bézier. Cliquez et maintenez le bouton de la souris enfoncé à l'endroit où vous voulez commencer à dessiner une nouvelle ligne, puis, tout en maintenant le bouton de la souris

enfoncé, faites glisser la souris dans la direction où vous voulez que la ligne se dirige. Cela fera apparaître les points de contrôle du point lui-même. Relâchez le bouton de la souris et cliquez à l'endroit où vous voulez placer le point suivant définissant la ligne. Faites à nouveau glisser la souris tout en maintenant le bouton enfoncé. Vous devriez commencer à voir comment la position des points de contrôle détermine la courbe de la ligne entre les points.

Dessin de courbes de Bézier : Notez les points de contrôle

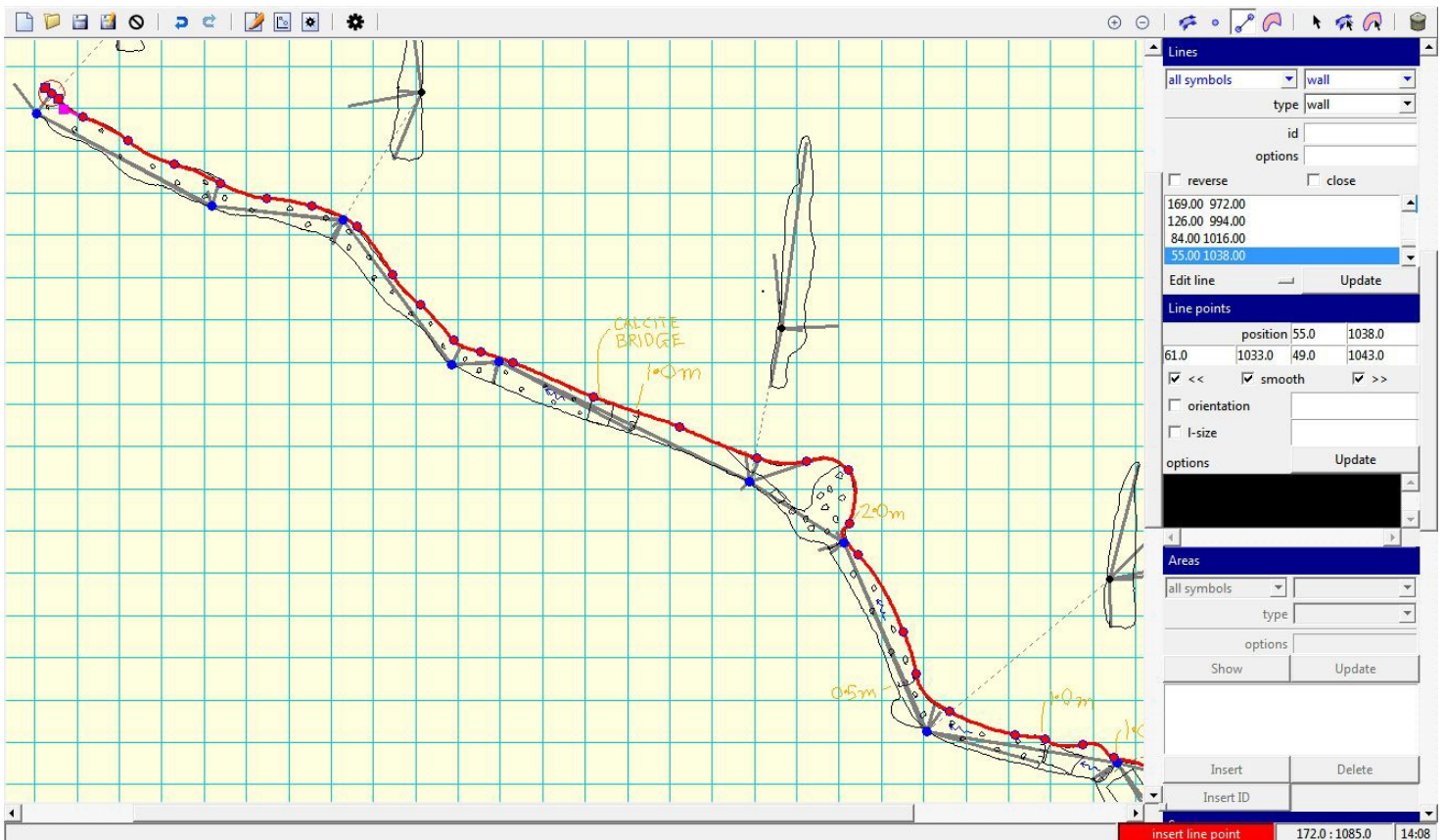


attachés à chaque point de la ligne, et la coche jaune indiquant l'"intérieur" de la grotte pour une ligne de paroi.

Répétez ce processus jusqu'à ce que la ligne entière soit définie. Appuyez ensuite sur la touche "Echap" pour quitter le mode "insertion de point de ligne". Vous pouvez maintenant sélectionner votre ligne avec la souris et la modifier.

Cliquez sur un point quelconque et faites-le glisser pour le repositionner. Les points de contrôle apparaissent pour chaque point sélectionné, et vous pouvez cliquer dessus pour les déplacer. Cette méthode de dessin de lignes courbes et lisses permet de contrôler finement l'apparence de la ligne. Entraînez-vous à dessiner quelques lignes et à les modifier. N'oubliez pas que vous pouvez annuler n'importe quelle étape en appuyant sur 'Ctrl+Z'. Vous pouvez également sélectionner une ligne entière et la supprimer en appuyant sur "Ctrl+D".

Une fois que vous vous êtes habitué à tracer des lignes, nous pouvons dessiner notre première paroi. En faisant le tour de notre dessin dans le sens inverse des aiguilles d'une montre, nous allons dessiner les parois en utilisant des lignes courbes. En commençant en bas à droite et en remontant le long du côté droit de notre croquis *PocketTopo*, placez des points de ligne pour dessiner une ligne courbe régulière correspondant au contour de la paroi. Lorsque vous arrivez en haut de l'esquisse, là où la galerie se poursuit hors de notre esquisse, nous devons commencer une nouvelle ligne. Terminez donc la ligne en appuyant sur la touche "Echap". Votre dessin devrait ressembler à quelque chose comme ceci.

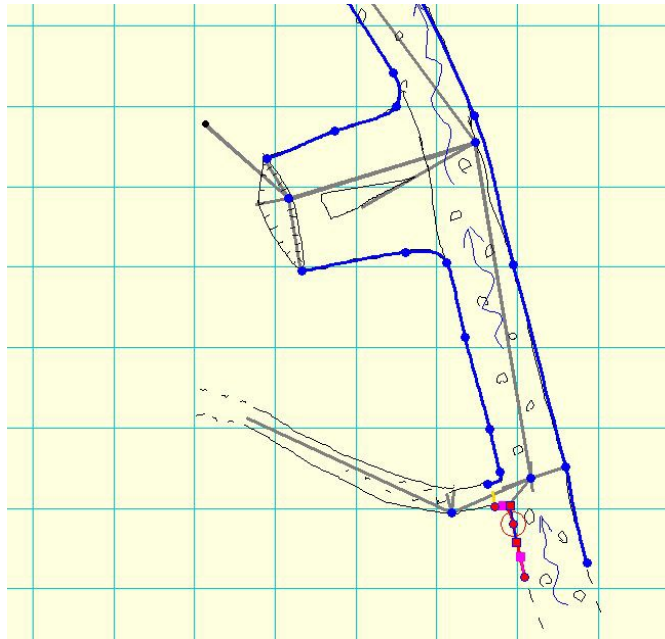


L'éditeur de cartes de *Therion* dessine tous les types de lignes avec la même apparence, nous devons donc vérifier que le type de ligne que nous avons dessiné est bien une ligne de paroi.

Assurez-vous que votre ligne est sélectionnée, puis consultez le panneau d'information "Lignes" sur la droite. Si le type n'est pas déjà défini comme étant une paroi, modifiez-le maintenant. Toutes les nouvelles lignes dessinées seront du type que vous avez sélectionné en dernier. Continuez à dessiner les parois. Vous pouvez commencer chaque nouvelle ligne en appuyant sur 'Ctrl+L'. Cela fait passer l'éditeur de carte en mode "insertion de ligne", ou termine une ligne si elle est déjà en mode "insertion de ligne" et le maintient dans ce mode, prêt à en dessiner une autre.

Lorsque vous revenez au début, quittez le mode "insertion de ligne" en appuyant sur la touche "Esc" pour terminer la dernière ligne. N'oubliez pas de cliquer sur "Enregistrer" lorsque vous avez fini de dessiner.

Pour cette leçon, vous devez dessiner les parois de la montée vers la galerie ci-dessus, mais laissez de côté la petite galerie latérale à l'extrémité sud de l'esquisse, car il n'a pas été entièrement inspecté lors de la séance topo que nous dessinons, et nous le dessinerons dans un autre morceau de claque (scrap) dans une leçon ultérieure. Vos parois devraient ressembler à quelque chose comme ceci.



Parois terminées pour cette leçon. Notez la coche jaune sur la ligne sélectionnée, qui indique de quel côté de la paroi se trouve "l'intérieur" de la galerie.

Voilà qui termine le dessin des parois pour cette leçon. Si vous n'êtes pas sûr de l'aspect du croquis à ce stade, trouvez le fichier `swildons.th2` dans le dossier Leçon 3 et ouvrez-le dans l'éditeur de cartes *Therion* pour voir à quoi notre dessin devrait ressembler. La dernière partie de cette leçon consiste à générer une topographie à partir de notre dessin.

De retour dans l'éditeur de texte (appuyez sur F1), nous devons inclure notre dessin dans notre projet. Ce n'est pas parce que nous avons créé un fichier dessin qu'il est automatiquement inclus dans notre projet *Therion*. Nous devons le faire en ajoutant une ligne juste en dessous de la ligne '`survey ...`' en haut du fichier texte `.th` pour inclure notre fichier `.th2`. Nous pouvons inclure des fichiers supplémentaires dans notre projet en utilisant la commande '`input`' suivie du nom du fichier. Ajoutez cette commande au début du bloc de données du relevé topo. Le début de notre fichier de projet devrait maintenant ressembler à ceci :

```
survey swildons input
swildons.th2
```

Maintenant que nous avons inclus notre fichier de dessin, *Therion* connaît les calques (scraps) qu'il contient. Mais nous devons encore spécifier comment ceux-ci doivent être utilisés dans le projet topo. Pour ce faire, nous définissons une carte. Une carte peut contenir soit des scraps, soit d'autres cartes (mais pas un mélange des deux). Nous allons définir une carte contenant notre scrap dans ce projet. Nous devons donner un nom à la carte. Comme il s'agit d'une carte en plan, la convention que nous recommandons est de lui donner le nom de la grotte, suivi de "M" pour carte (NDLR : M = MAP, en français on pourrait mettre C = CARTE) et de "P" pour plan. Nous allons donc nommer notre carte "swildonsMP" dans cet exemple. À l'intérieur de la définition de la carte, nous mettons le nom de notre scrap. Donc directement sous la ligne '`input...`' nous ajoutons ce qui suit :

```
Map
```

```
swildonsMP swilSP1
```

```
endmap
```

Enfin, nous devons modifier le fichier de configuration pour lui dire de créer une sortie pour cette carte. Passez à la fenêtre de compilation (F3), et ajoutez la ligne suivante au fichier de configuration :

```
export map -projection plan -o swildons.pdf
```

Nous donnons ici l'instruction de créer un document pdf pour notre carte. Comme nous n'avons qu'une seule carte dans le projet, nous n'avons pas besoin de préciser le nom de la carte. Nous indiquons simplement qu'il s'agit d'un plan, et le nom de fichier que nous voulons donner au fichier de sortie. Compilez le projet (appuyez sur F9) et si tout a été fait correctement, vous devriez voir la barre verte 'OK' et constater qu'un fichier 'swildons.pdf' a été généré. Ouvrez-le pour visualiser votre plan topo *Therion* ! Vous devriez voir un dessin qui inclut tout ce que nous avons dessiné jusqu'à présent, les murs, les stations topo et *Therion* montre également les visées de cheminement qu'il connaît à partir des données du relevé. Notez qu'il inclut également certaines des informations que nous avons saisies dans les données, comme la date et les membres de l'équipe qui ont effectué les relevés.

Avant de passer à d'autres dessins, permettez-moi de partager avec vous une astuce précieuse. Le visualisateur de pdf par défaut des ordinateurs *Windows* verrouille tout fichier pdf ouvert, ce qui empêche *Therion* de pouvoir les écraser. Essayez de compiler votre projet *Therion* une seconde fois sans fermer le visualiseur pdf. Vous obtiendrez une erreur, et la barre d'état du compilateur affichera " ERROR " en rouge. Regardez la sortie du journal et vous verrez l'erreur :

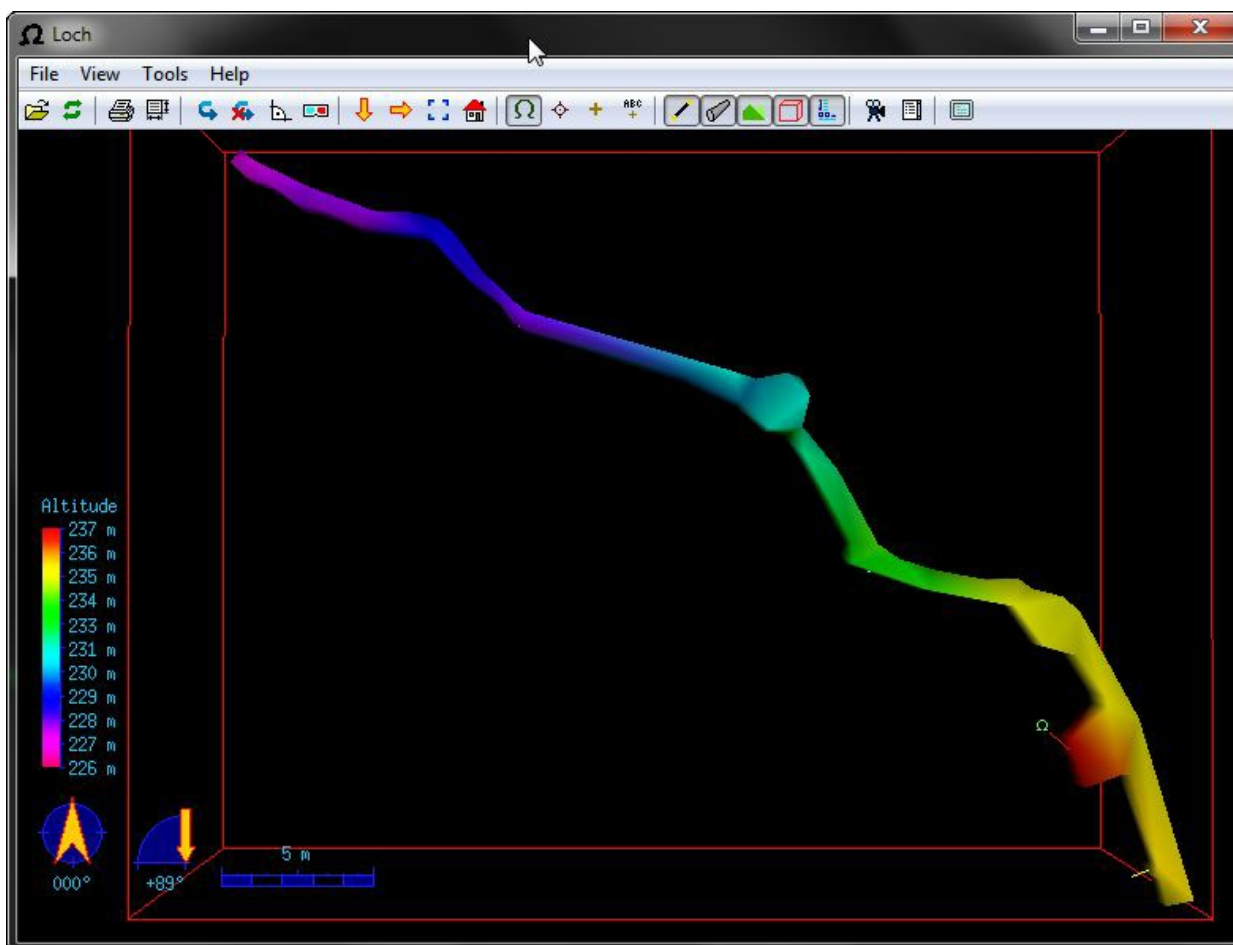
```
therion.exe: error -- cp exit code -- 1
```

Si vous utilisez la visionneuse PDF par défaut, vous verrez souvent cette erreur, et c'est une douleur de devoir se rappeler de fermer le fichier PDF avant chaque compilation. Il se peut que vous obteniez toujours cette erreur après avoir fermé le visualiseur de PDF. Cela est généralement dû au fait que l'Explorateur Windows génère un aperçu du fichier PDF pour le fichier sélectionné, et qu'il utilise le visualiseur de PDF en arrière-plan pour le faire. Votre fichier est donc toujours verrouillé. Assurez-vous donc que votre fichier PDF n'est pas sélectionné dans l'Explorateur Windows et qu'il n'est pas ouvert dans le lecteur de PDF. Maintenant, vous devriez être en mesure de compiler votre projet à nouveau. Pour éviter de devoir fermer et rouvrir les fichiers PDF à chaque fois que vous voulez compiler, je vous recommande vivement d'utiliser l'application gratuite *Sumatra PDF viewer*. Le lien vers cette visionneuse se trouve dans la section des liens logiciels au début de ce tutoriel. Une fois que vous avez *Sumatra PDF viewer* sur votre ordinateur, au lieu de double-cliquer pour ouvrir votre document PDF, cliquez dessus avec le bouton droit de la souris et dans le menu contextuel, sélectionnez le sous-menu "Ouvrir avec...". Sélectionnez "Choisir le programme par défaut" et accédez au fichier SumatraPDF.exe sur votre ordinateur. Vous n'avez pas besoin d'en faire le programme par défaut si vous ne le souhaitez pas, car une fois que vous avez ouvert un pdf à partir de ce menu, vous devriez constater qu'à l'avenir, *Sumatra PDF* apparaîtra dans le menu contextuel "Ouvrir avec...". Lorsque votre plan d'arpentage pdf est ouvert dans la visionneuse *Sumatra PDF*, vous pouvez recompiler votre projet *Therion* quand vous le souhaitez,

et le pdf se rafraîchira automatiquement dans la visionneuse. Cela facilite grandement le dessin et la vérification du résultat. Mieux encore, si vous avez plusieurs moniteurs, vous pouvez garder la visionneuse PDF ouverte sur un écran et l'éditeur *Therion* sur l'autre, ce qui vous facilite grandement la tâche lorsque vous dessinez vos topographies.

Il y a une autre chose qui a changé dans notre résultat maintenant que nous avons ajouté une carte. Le modèle 3D de la grotte utilisera désormais les parois de la galerie que nous avons dessinées pour générer un meilleur modèle.

Ouvrez le fichier `swildons.lox` et jetez-y un œil. Maintenant, au lieu d'afficher de fins cylindres le long des visées de cheminement, vous devriez voir une bien meilleure représentation 3D de la galerie de la grotte que nous avons dessinée.



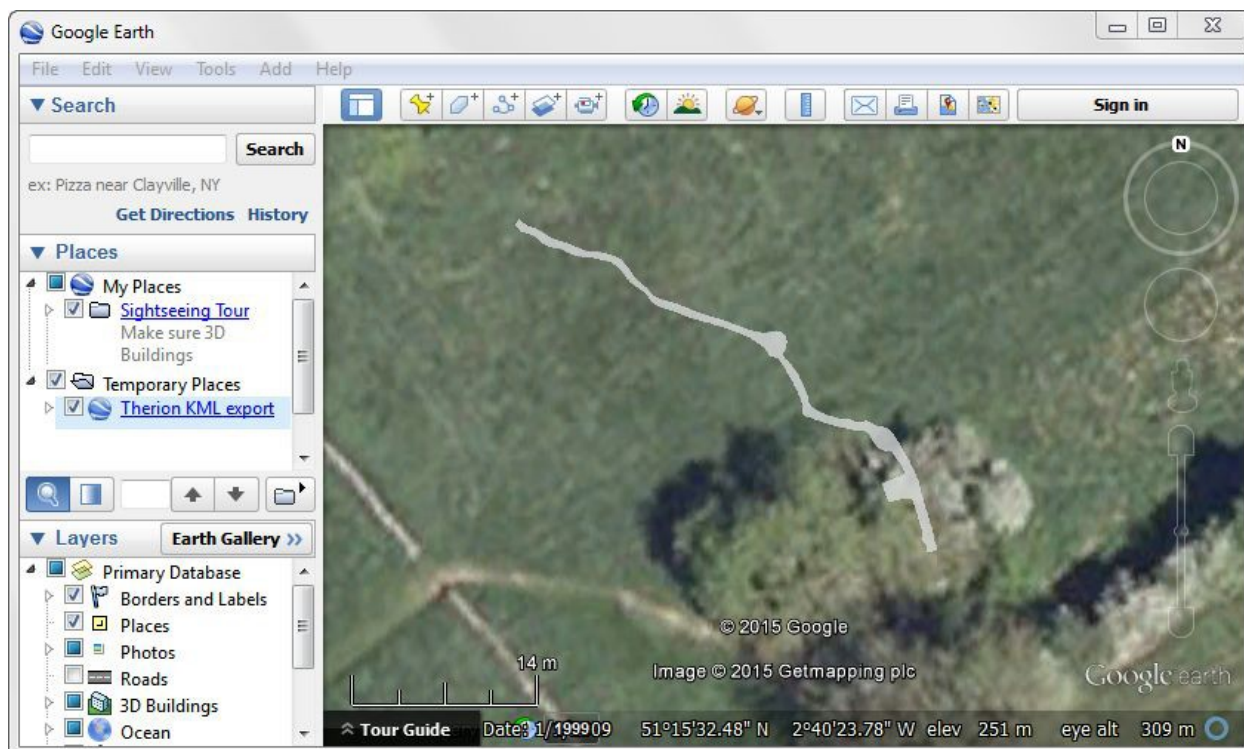
Maintenant que nous avons dessiné quelques parois et inclus un scrap dans notre projet, nous pouvons générer d'autres types de cartes. Passez à la fenêtre de compilation (F3), et ajoutez la ligne suivante au fichier de configuration :

```
export map -proj plan -fmt kml -o swildons_map.kml
```

Notez maintenant que cela diffère de la commande d'exportation précédente que nous avons utilisée pour une sortie KML. Il s'agissait d'un modèle (qui générait un modèle kml de ligne centrale à partir des



données du relevé topo), alors que cette commande est une exportation de carte. Nous spécifions également le type de carte (plan). Si vous comparez cette commande à la dernière commande d'exportation de carte plan, vous pouvez constater que nous avons utilisé l'option "projection plan", alors qu'ici nous l'avons raccourcie en "proj plan". Les deux formes font la même chose. D'autres options ont également des formes longues et courtes (par exemple, nous utilisons 'o' au lieu de la forme complète 'output' pour spécifier le nom du fichier de sortie, et 'fmt' au lieu de 'format'). Vous pouvez utiliser ce que vous préférez. Vous trouverez tous les détails dans le livre de *Therion* (recherchez 'export' pour trouver toutes les options que *Therion* supporte). Compilez le projet et vous devriez voir un nouveau fichier de sortie 'swildons\_map.kml'. Ouvrez-le pour voir comment la carte kml diffère dans *Google Earth* du modèle de ligne centrale de cheminement que nous avons produit précédemment.



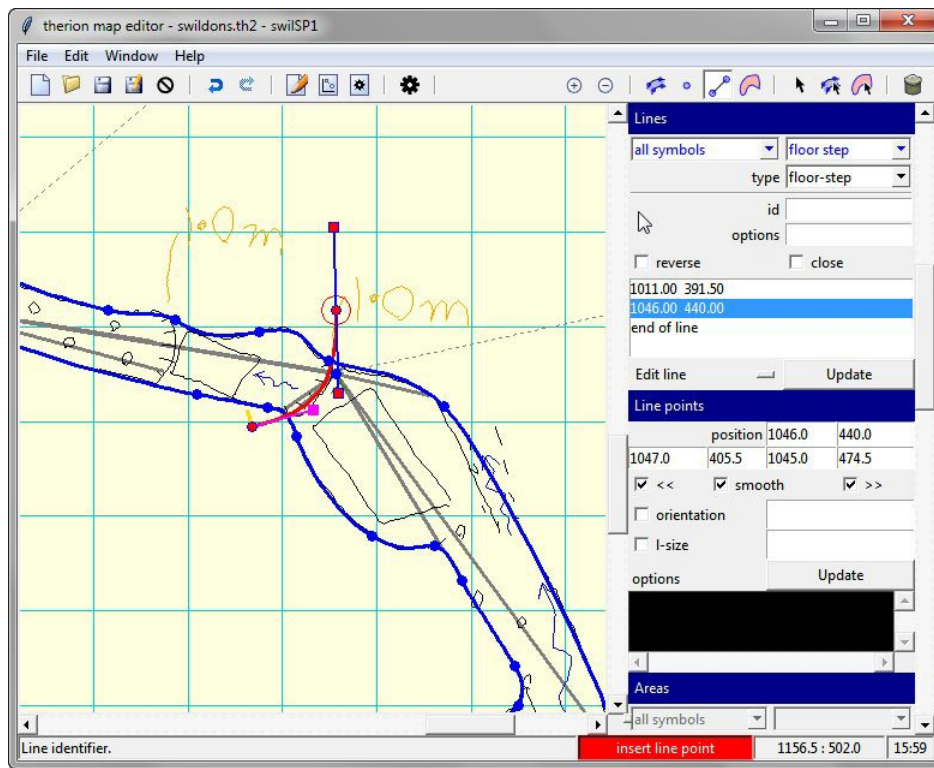
(Capture d'écran de *GoogleEarth*. Le Copyright de l'image est visible)

Pour accélérer la compilation de notre projet pendant que nous travaillons sur notre dessin, nous pouvons commenter certaines de ces commandes d'exportation dans le fichier *thconfig*. Placez un caractère '#' devant la commande d'exportation pour empêcher l'exécution de cette commande lors de la compilation du projet. *Therion* ignorera tout texte sur une ligne suivant un '#'. Faites cela pour toutes les commandes d'exportation, à l'exception de la carte pdf. Nous les activerons à nouveau dans les leçons suivantes, lorsque nous aurons dessiné une plus grande partie de la grotte.

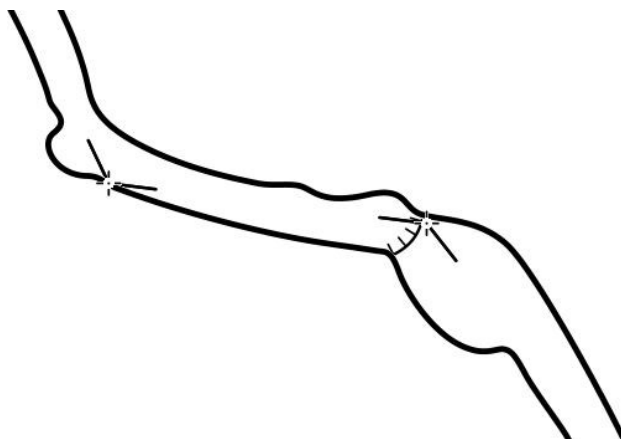
## Leçon 4: Dessiner des symboles en utilisant d'autres types de lignes

Dans la leçon précédente, nous avons appris à dessiner des lignes et à vérifier le type de ligne. Maintenant que les parois principales de notre étude sont dessinées, nous devons ajouter quelques détails. La galerie de notre grotte comporte quelques petites montées, nous allons donc commencer par ajouter quelques lignes de marches au sol (ressaut). *Therion* coupera les lignes et les symboles de façon à ce qu'ils n'apparaissent que dans les galeries de la grotte, nous n'avons donc pas besoin de commencer et de

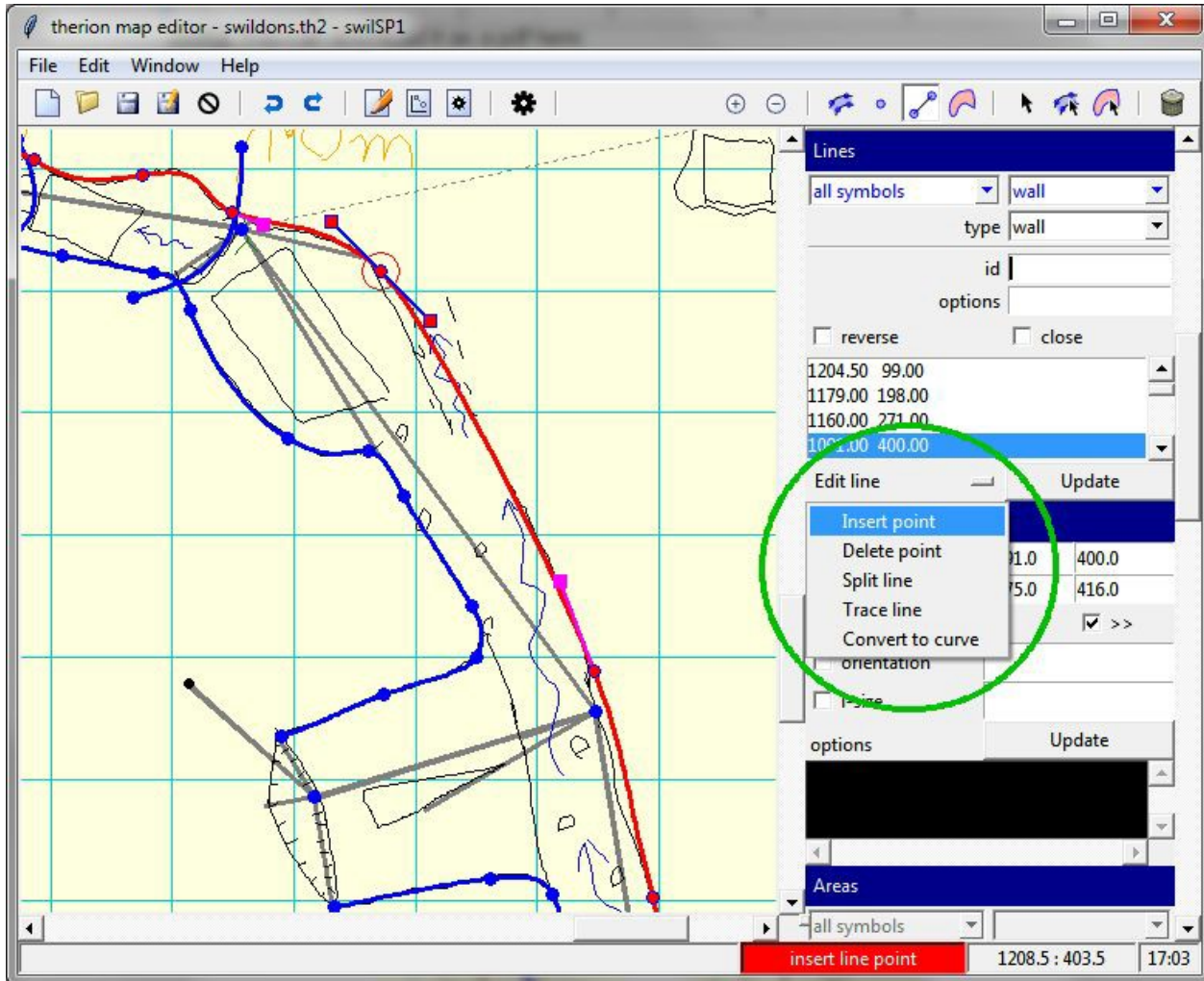
terminer ces lignes parfaitement sur les parois de la galerie (en gros vous pouvez déborder). Commencez une nouvelle ligne, et cliquez un peu à l'extérieur des parois de la galerie pour la démarrer. Dessinez la nouvelle ligne le long de la ligne du sol sur le croquis *PocketTopo* comme indiqué ci-dessous. Lorsque la ligne est terminée, définissez le type de ligne sur "floortep" dans le panneau d'information des lignes.



Notez la direction de la coche jaune au début de la ligne. Elle indique de quel côté les marques le long de la ligne seront dessinées lors du rendu. Le symbole de l'ISU pour les marques de ressaut est une ligne avec des tics pointant vers le côté de la ligne où le plancher est le plus bas, donc notre coche pointe dans la bonne direction. Si ce n'était pas le cas, nous pourrions inverser la direction de la ligne en cochant la case "reverse" dans le panneau d'information des lignes. Compilez à nouveau le projet et visualisez le PDF. Vous devriez voir la ligne de ressaut ressembler à ceci. (Remarquez que la ligne n'est dessinée qu'à l'intérieur de la galerie).



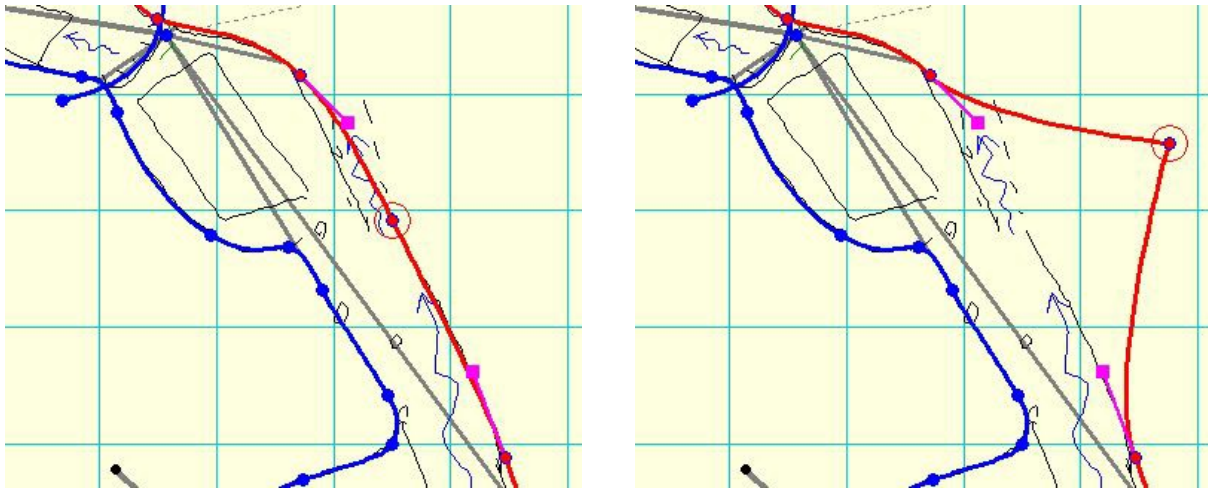
Une fois que nous avons défini un type de ligne, toutes les nouvelles lignes que nous traçons seront du même type par défaut. Il est donc logique de dessiner toutes les lignes des marches du plancher ensemble, avant de passer à d'autres types de lignes. Tracez les lignes d'escalier pour chaque étape de la galerie. Si vous n'êtes pas familier avec les symboles et les styles de lignes utilisés dans ces croquis, le document de référence des symboles d'arpentage de l'ISU vous sera utile. Vous pouvez le télécharger en format pdf ici : [http://www.carto.net/neumann/caving/cavesymbols/uis\\_signatures\\_english.pdf](http://www.carto.net/neumann/caving/cavesymbols/uis_signatures_english.pdf)



Ensuite, nous allons examiner l'endroit où l'eau se perd dans les galets sous la paroi. Dans la leçon précédente, nous avons simplement dessiné une paroi continue après ce point, mais nous pouvons ajouter plus de détails ici. Tout d'abord, nous devons briser la ligne de paroi afin de pouvoir dessiner la perte du ruisseau ici. Il n'y a pas de point sur la ligne à l'endroit où nous voudrions idéalement la couper, donc nous allons d'abord insérer un point de ligne, et ensuite séparer la ligne en deux à ce point. Sélectionnez le point de la ligne juste après l'endroit où nous voulons insérer un nouveau point (la ligne doit être représentée en rouge avec le point sélectionné indiqué par un cercle rouge autour de lui).

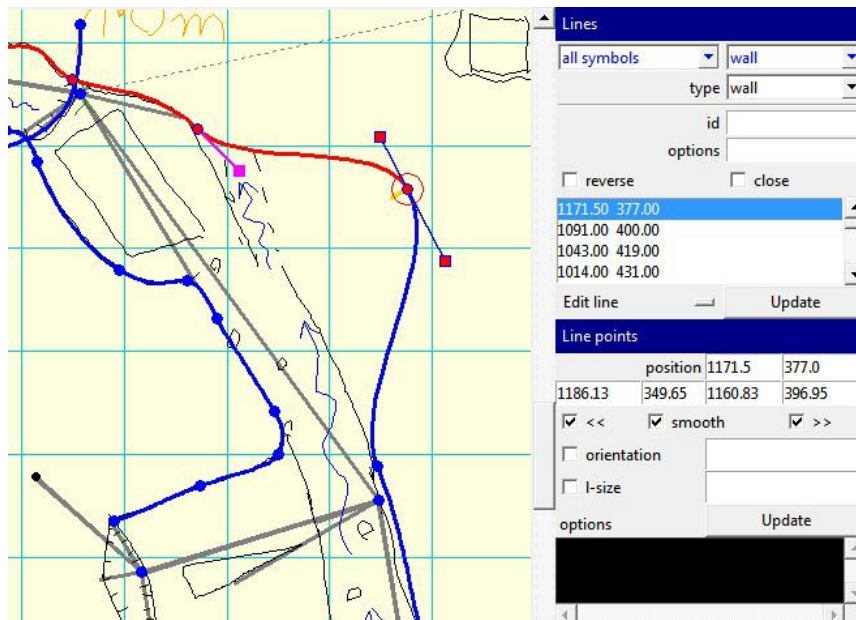
Cliquez maintenant sur "Editer la ligne" dans le panneau Lignes pour ouvrir le menu d'édition de ligne (entouré en vert ici) et sélectionnez l'élément "Insérer un point". Vous pouvez maintenant cliquer sur la

ligne pour insérer un nouveau point, comme illustré ici.



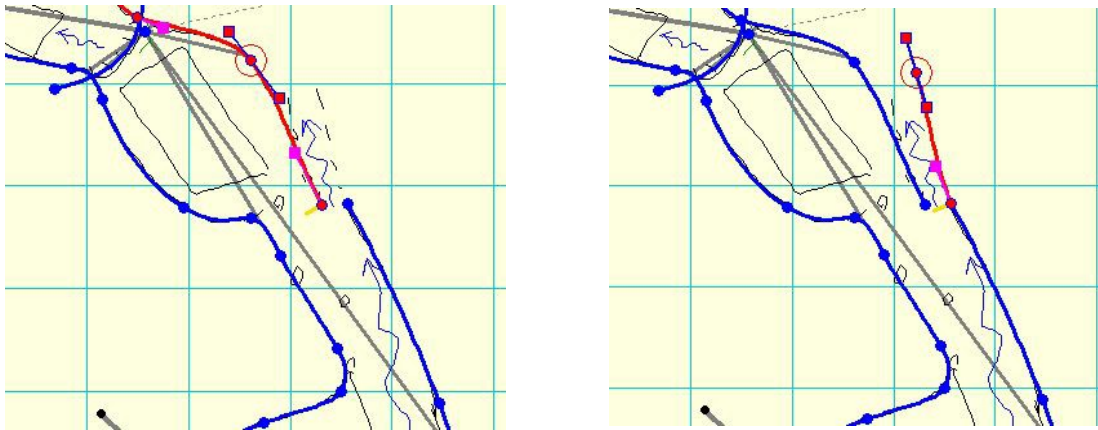
Une fois le point inséré, n'oubliez pas d'appuyer sur la touche Echap pour quitter le mode "Insertion de point", ou tout autre clic sur le dessin insérera d'autres points. Notez que le point n'affiche pas les poignées de la courbe de Bézier. Ainsi, la ligne présentera une courbure prononcée à cet endroit si nous déplaçons le point sur le côté, comme le montre la deuxième image ci-dessus. Nous pouvons ajouter les points de contrôle en cochant la case "smooth" dans le panneau de la ligne avec le point toujours sélectionné. En gardant le point sélectionné, cliquez maintenant sur le menu "Edit line" et choisissez "Split line". Vous devriez maintenant avoir deux lignes séparées, comme vous le voyez ci-dessous. Seule une des lignes peut être sélectionnée, l'une est donc redevenue bleue, et l'autre rouge.

Vous pouvez également voir la coche jaune à l'extrémité de la nouvelle ligne, car il s'agit maintenant du premier point de cette ligne.



Maintenant que la ligne a été divisée, nous pouvons déplacer les points d'extrémité des deux lignes dans les positions que nous voulons. Ensuite, nous pouvons dessiner une nouvelle ligne de paroi pour la perte

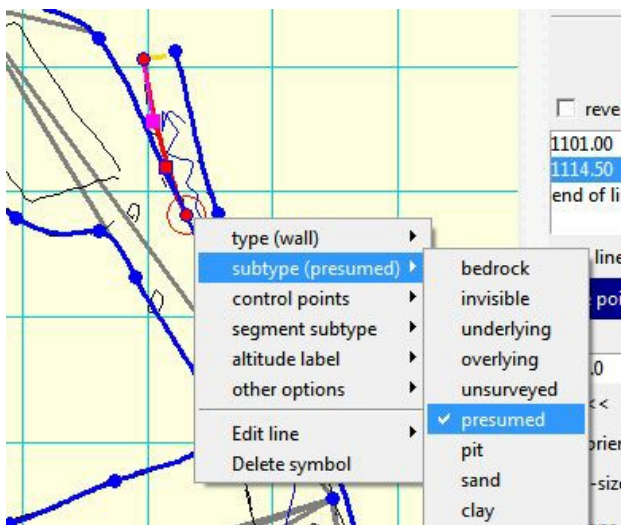
du ruisseau. Notez le côté où la coche jaune apparaît (elle doit pointer vers la galerie).



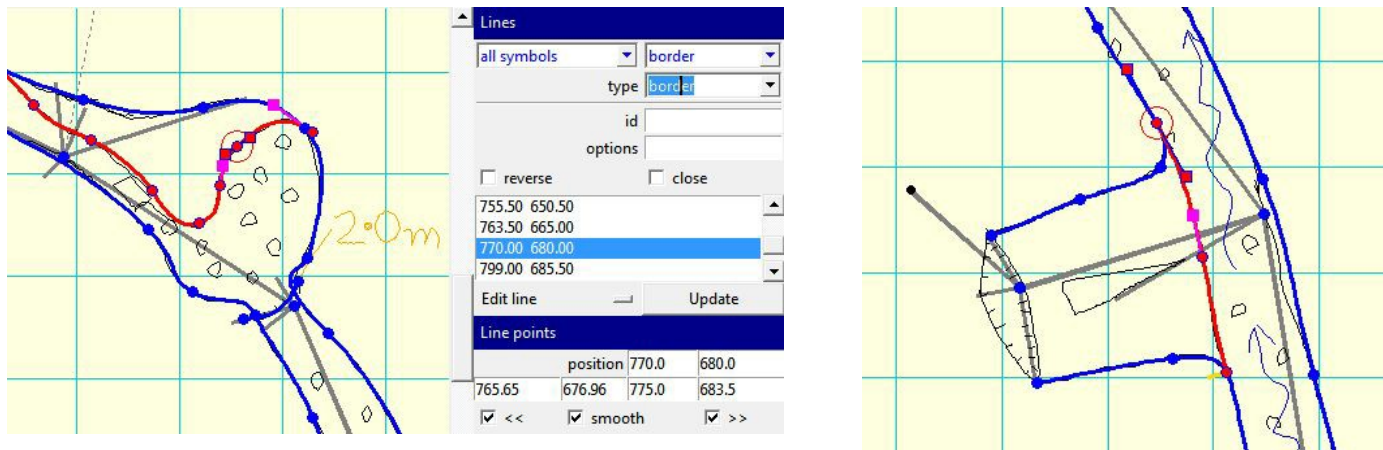
Cette nouvelle ligne de paroi fait partie d'une petite galerie annexe que nous n'avons pas encore topographiée, nous devons donc l'indiquer en changeant le type de ligne. Nous voulons toujours qu'il s'agisse d'un type de paroi, car elle fait partie du contour de la cavité (elle devrait former une paroi dans le modèle 3D). Mais nous pouvons ajouter un sous-type en entrant le texte suivant dans la boîte d'options.

```
-subtype presumed
```

Cette méthode d'application d'options supplémentaires s'applique à de nombreux objets dans les dessins *Therion*. Maintenant que nous avons défini une option, la ligne suivante que nous dessinons reprendra également le même réglage. Il est donc recommandé de définir les options sur la première ligne que nous dessinons, puis de dessiner le reste des lignes de ce type. Sinon, vous finirez par avoir dessiné plusieurs lignes et devrez revenir en arrière pour définir les options sur chacune d'entre elles. Heureusement, nous n'avons pas besoin de nous souvenir de tous les codes d'option que *Therion* prend en charge, car ils peuvent être listés à l'aide des menus de clic droit. Cliquez avec le bouton droit de la souris sur la nouvelle ligne et vous verrez que le menu contextuel vous permet de définir le sous-type. Dessinez la paroi présumé de l'autre côté de la perte, et fabriquez le pdf. L'esquisse et le dessin rendu final devraient ressembler à ce qui suit.



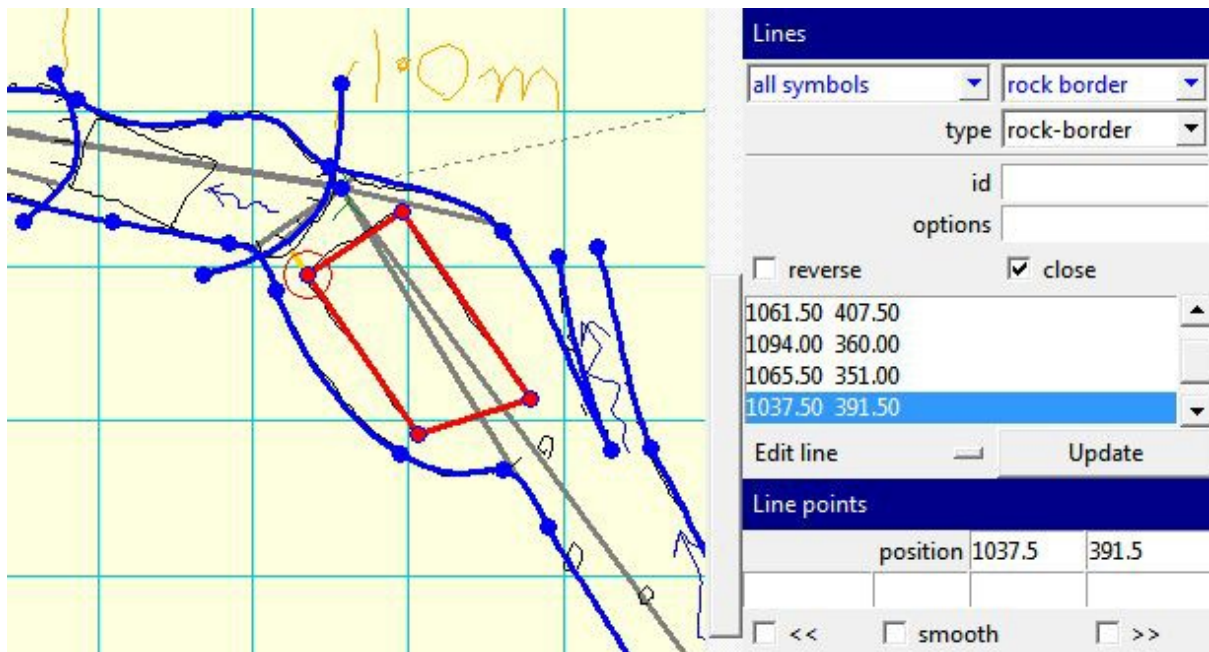
Pour des dénivelés moins importants de la hauteur du sol, nous pouvons utiliser le type de ligne "bordure". Nous avons quelques endroits dans notre galerie où nous voulons dessiner une simple ligne de bordure. Tracez une ligne qui correspond à la ligne courbe du croquis. Nous voulons également dessiner une ligne de bordure en bas de la descente (une montée appelée "échelle de Jacob" pour ceux qui connaissent la grotte). Voici à quoi devraient ressembler ces deux lignes.



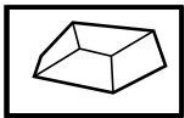
Il importe peu de savoir de quel côté de la ligne de bordure se trouve la coche jaune, car ce type de ligne n'a pas de côtés différents.

La galerie de la grotte contient quelques rochers que nous voulons dessiner à des endroits spécifiques (nous verrons plus tard comment faire en sorte que *Therion* dessine des rochers pour indiquer un sol couvert de rochers, mais ici nous voulons dessiner l'emplacement exact de ces rochers dans la galerie). Pour cela, nous utilisons le type de ligne 'rockborder'. Ces lignes diffèrent du type de ligne 'bordure' de base en ce sens que lorsque nous dessinons une ligne fermée (une ligne qui commence et se termine au même point) en utilisant le type de ligne 'bordure rocheuse', *Therion* rendra la forme avec un remplissage opaque, ce qui signifie que les rochers dessinés au-dessus des remplissages de la galerie ne montreront pas le remplissage à travers la forme. Par exemple, un rocher dessiné dans une zone d'eau sera rendu au-dessus de l'eau, plutôt que comme de simples lignes dans l'eau. Il y a une leçon entière à venir sur les zones et les remplissages de symboles, donc nous détaillerons cela plus tard.

Pour dessiner un rocher, nous voulons des coins nets aux angles. Ainsi, au lieu d'utiliser la méthode du cliquer-glisser pour faire sortir les poignées de contrôle de la courbe, nous cliquons et relâchons la souris. Commencez une nouvelle ligne (Ctrl+L) et cliquez sur les quatre coins du grand bloc rocheux dans la galerie. Terminez le rocher en cliquant à nouveau sur le premier point où nous avons commencé la ligne. Remarquez que l'éditeur de *Therion* quitte le mode "insertion de point de ligne" lorsque nous fermons une ligne. Pour dessiner le deuxième bloc, nous devons donc appuyer à nouveau sur Ctrl+L (ou cliquer sur l'icône de la nouvelle ligne dans la barre d'outils), mais n'oubliez pas de définir le type de ligne du premier bloc sur 'rockborder'. De cette façon, la prochaine ligne que nous dessinerons sera déjà du type correct et nous n'aurons pas besoin de revenir en arrière pour définir le type correct pour tous les rochers que nous avons dessinés. Notez également que dans le panneau de la ligne, la case à cocher "close" est cochée. Cela indique que nous voulons que ce bloc soit traité comme une zone fermée (il couvrira donc tout symbole rempli en dessous de lui). *Therion* ferme automatiquement la ligne lorsque nous cliquons sur le point de départ pour terminer la ligne.



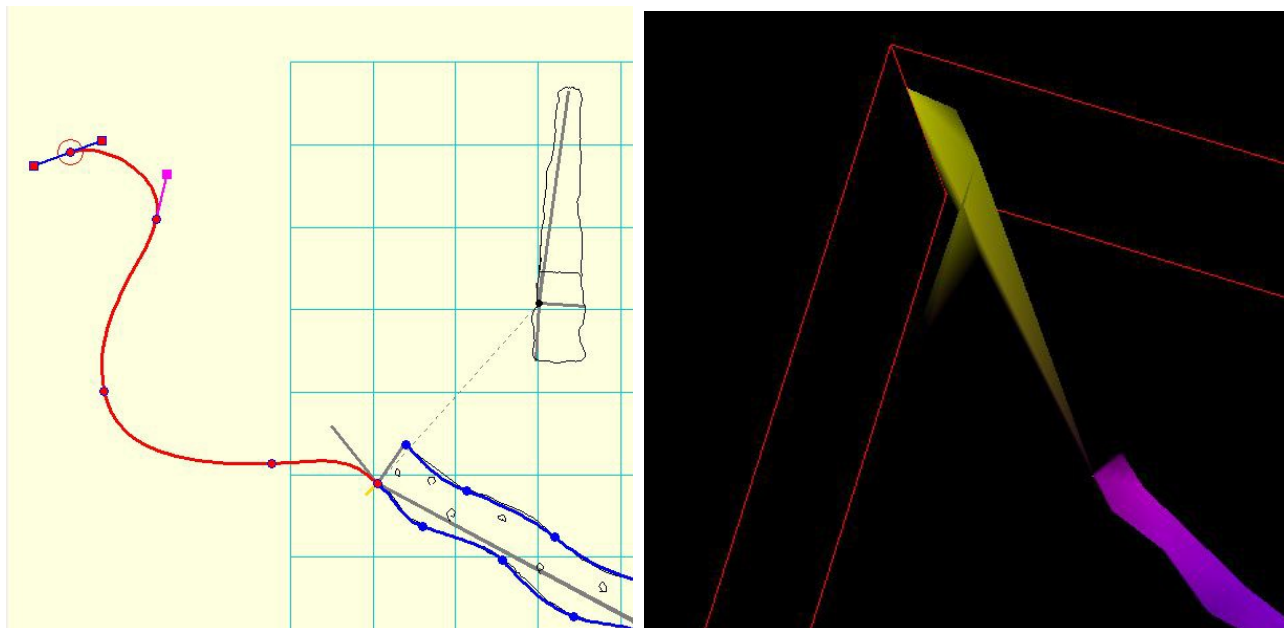
Le bloc rocheux est dessiné avec une ligne de type "rockborder".  
 La ligne est fermée (la case de fermeture est cochée) et les points  
 de la ligne ne sont pas lisses (la case lisse n'est pas cochée).



Vous avez peut-être remarqué qu'il existe également un type de ligne "rockedge" dans l'éditeur *Therion*. Il s'agit d'une ligne légèrement plus fine utilisée pour dessiner les détails à l'intérieur des rochers. Si vous aimez ce genre de choses, utilisez la ligne rockborder pour la ligne extérieure autour de votre rocher, et la ligne rockedge pour les lignes intérieures.

Personnellement, je préfère simplement dessiner le contour de mes rochers. J'ai assez de relevés à faire sans avoir à dessiner chaque bloc à la main ! Mais votre projet peut être différent et les blocs ont un bel aspect si vous dessinez une étude à grande échelle d'une petite grotte. Pour ce tutoriel, dans les fichiers d'exemple, j'ai juste dessiné le contour des rochers restants dans la galerie en utilisant le type de ligne rockborder.

Nous allons terminer cette leçon sur les types de lignes avec quelques aspects importants des parois et du contour de notre modèle de grotte, car vous rencontrerez inévitablement certains de ces problèmes assez vite lorsque vous commencerez à dessiner vos propres relevés de grottes. Notre exemple de galerie souterraine ne contient pas les éléments nécessaires pour illustrer cette dernière partie de la leçon, nous allons donc les dessiner à la fin de la galerie, juste pour couvrir ces points importants. Commencez par étendre la paroi de la galerie comme indiqué ici. Nous devons également générer le modèle lox à nouveau, donc décommentez la commande « export model » dans le fichier de configuration pour générer le fichier lox et compiler le projet.

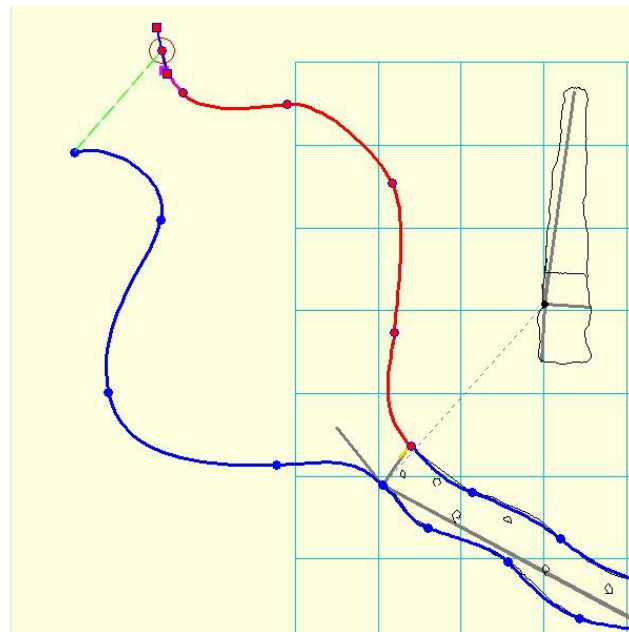
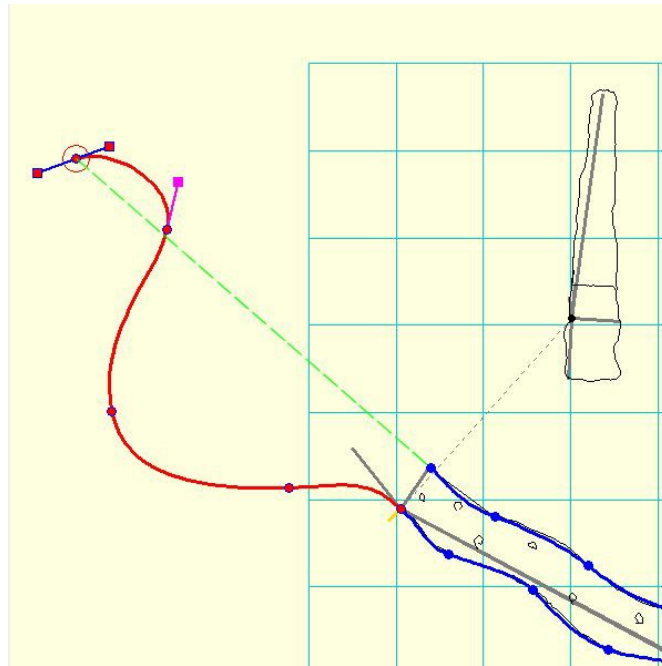


Le modèle résultant dans le *Loch viewer* montre des choses étranges avec les murs, comme indiqué ci-dessus. Il y a deux problèmes avec la ligne de paroi telle qu'elle est dessinée. Premièrement, la coche jaune pointe vers l'extérieur de la galerie, ce qui fait que les objets de *Therion* à l'extérieur de la grotte se trouvent à droite de cette paroi. Nous pouvons facilement résoudre ce problème en cochant la case 'Reverse' dans le panneau Lines. Cependant, cela ne résout pas les problèmes d'affichage dans le modèle lox. L'autre problème est que le contour de notre grotte se croise avec lui-même. Si vous regardez la fenêtre du compilateur, vous verrez que la barre d'état de la compilation de *Therion* n'est plus verte. Elle est devenue orange et indique 'Warning'. Regardez dans la sortie de *Therion* dans le volet inférieur de la fenêtre du compilateur, ou si vous ne trouvez pas l'avertissement parmi tous les autres textes, ouvrez le fichier *therion.log* que vous trouverez au même endroit que vos fichiers de projet et cherchez le mot 'warning'. Vous devriez trouver ce qui suit :

```
therion.exe: warning -- swilSP1@swildons -- invalid scrap outline
```

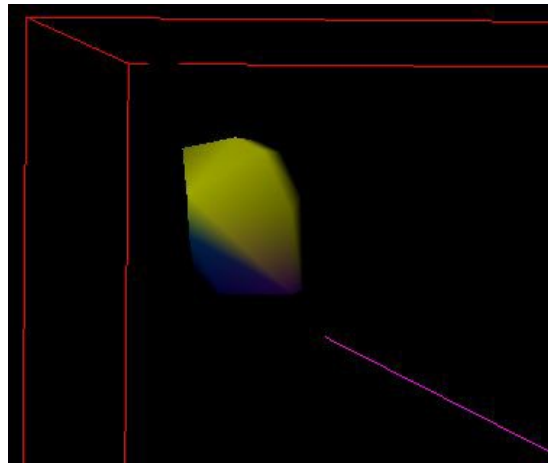
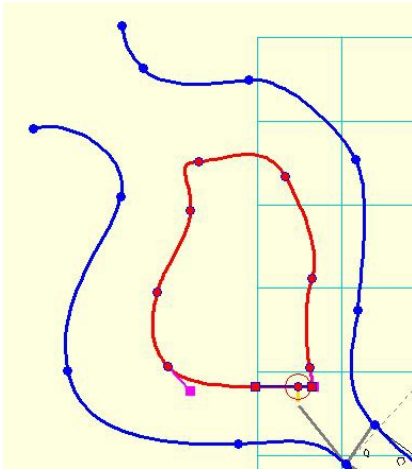
*Therion* nous dit quel est le problème, et quel objet du projet contient le problème. C'est ici que nous voyons l'avantage d'utiliser une convention de nommage pour nos objets. Nous pouvons dire d'après le nom qu'il s'agit d'un scrap de plan (parce que nous utilisons SP dans le nom de nos scraps de plan), donc nous savons que dans notre topo appelée *swildons*, il y a un scrap nommé *swilSP1* qui a un contour invalide. Cela se produit généralement lorsque vous avez dessiné une ligne destinée à être un type de symbole (comme une ligne de bordure) mais que vous l'avez laissée de type "paroi" par erreur. Dans ce cas, c'est parce que si vous tracez une ligne droite entre les extrémités de vos parois, cette ligne traverse la paroi. En d'autres termes, le contour du scrap se croise lui-même. Les illustrations suivantes montrent comment *Therion* complète le contour avant et après que nous l'ayons fixé avec une autre paroi. (La ligne verte pointillée montre ce que *Therion* traitera comme étant le contour du scrap. Vous ne verrez pas cette ligne verte dans l'éditeur, je l'ai dessinée au-dessus de la capture d'écran pour illustrer ce point).





Vous pouvez voir comment le contour du morceau de calque (scrap) traverse la ligne de paroi lorsque *Therion* rejoint les extrémités de la ligne de paroi sur le croquis de gauche. En dessinant la paroi opposée à l'endroit indiqué, nous pouvons corriger le problème de façon à ce que le contour qui traverse les extrémités de la ligne de paroi ne croise plus une paroi. Maintenant, le contour devrait être valide, le statut du compilateur devrait à nouveau afficher un OK vert et le modèle dans le *Loch viewer* devrait montrer des parois correctes comme prévu.

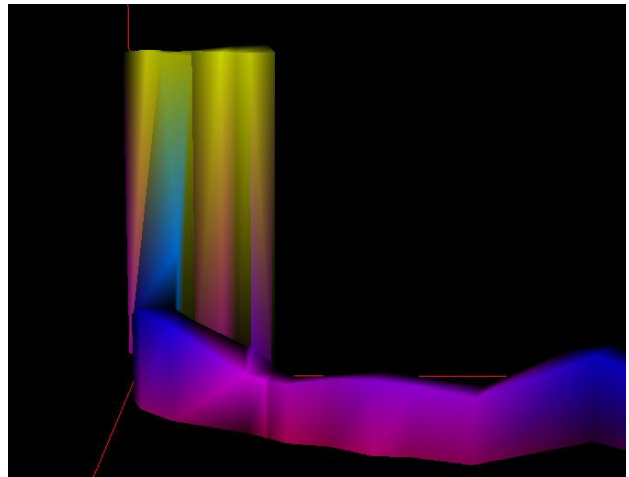
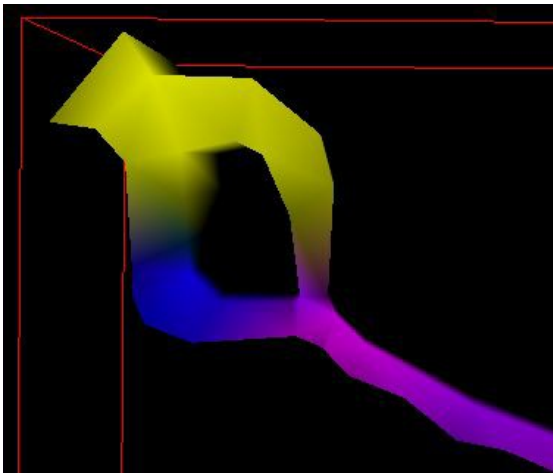
Nous arrivons maintenant à l'autre cas courant qui déroute tous les nouveaux venus dans *Therion*. Les piliers. Imaginez que l'élargissement de la galerie que nous avons dessiné est en fait l'endroit où la galerie se sépare en deux et rejoint l'autre côté d'un solide pilier rocheux. Nous pourrions dessiner le pilier avec une ligne de bordure, mais il s'agit en fait d'une paroi. Il devrait donc s'agir d'une ligne de paroi. Dessinez-le comme indiqué ici.



Le compilateur de *Therion* affiche à nouveau un avertissement et le modèle de *Loch* montre maintenant l'intérieur du pilier mais plus aucune des parois extérieures de la galerie.

```
warning -- swilSP1@swildons -- multiple scrap outer outlines not supported yet
```

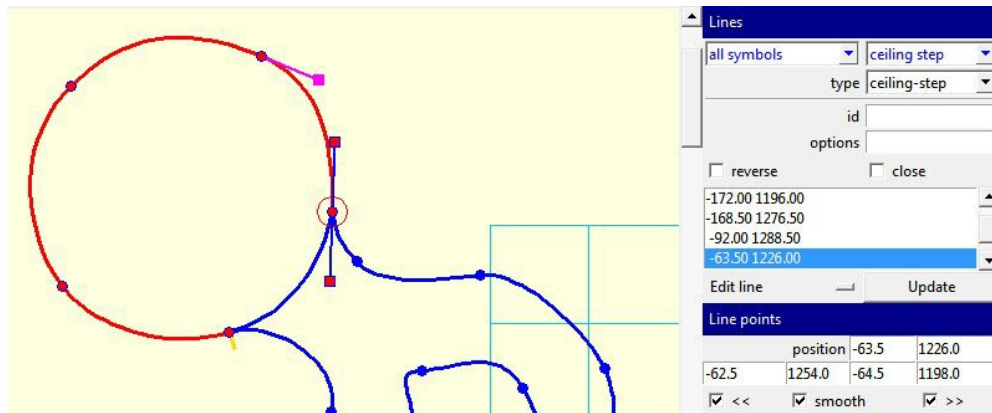
Nous devons dire à *Therion* que cette nouvelle ligne de paroi est entièrement à l'intérieur du contour de la galerie. Nous utilisons pour cela l'option '`outline in`'. Nous pouvons soit entrer le texte '`outline in`' dans le champ des options de ligne dans le panneau des lignes, soit cliquer avec le bouton droit de la souris sur la ligne de paroi du pilier et sélectionner dans le menu popup '`other options/outline/in`' qui définira la même option pour nous sans que nous ayons besoin de nous souvenir du texte exact de l'option. Faites un nouveau rendu du projet et cette fois tout devrait bien se passer.



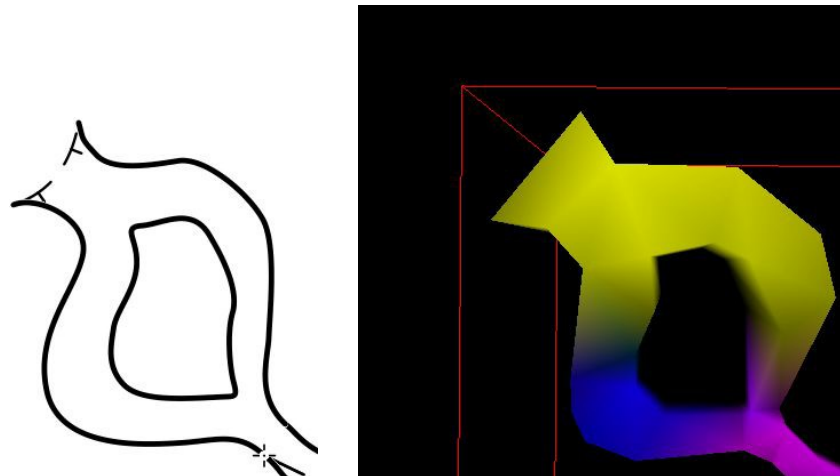
Nous voyons maintenant pourquoi nous devrions utiliser un type de ligne de paroi pour le pilier, et non une ligne de bordure. Le modèle montre maintenant deux passages distincts comme c'est le cas dans la galerie de la cavité que nous avons dessinée, plutôt qu'une salle qu'il aurait modélisée si nous n'avions dessiné que les parois extérieures en utilisant des lignes de paroi. Au moins la vue en plan du modèle semble correcte. *Therion* a basé les largeurs des galeries sur les parois que nous avons dessinées dans notre croquis en plan (image de gauche ci-dessus). Nous n'avons pas encore donné à *Therion* suffisamment d'informations sur les hauteurs des galeries. Ainsi, dans l'image de droite, vous pouvez voir que l'aspect du modèle n'est pas très bonne. *Therion* a fait une estimation de la hauteur des galeries et s'est trompé. Un

côté de la boucle autour du pilier a un plafond bas et l'autre un plafond très haut. Les hauteurs de galerie du modèle *lox* sont interprétées par *Therion* à partir des données du relevé topo et ne donnent pas toujours un bon résultat. Il n'est pas parfait sur la majeure partie de la zone, mais il semble assez correct comme modèle pour que nous puissions nous en accommoder. Dans une prochaine leçon, nous verrons comment nous pouvons le modifier pour obtenir un meilleur modèle.

Enfin, dans cette leçon, nous examinerons quelques options supplémentaires qui peuvent être appliquées aux lignes autres que de parois pour modifier la façon dont elles sont traitées. Imaginez que notre galerie se termine par une grande cheminée. Nous voulons dessiner cette cheminée sur notre plan. Pour indiquer une augmentation de la hauteur du plafond, nous utilisons le type de ligne "ceilingstep". Utilisez ce type de ligne pour dessiner un bel orifice rond à la fin de la galerie. Notez que nous avons utilisé deux types de lignes, une à travers l'extrémité de la galerie, et une autre autour des parois extérieures de l'aven.

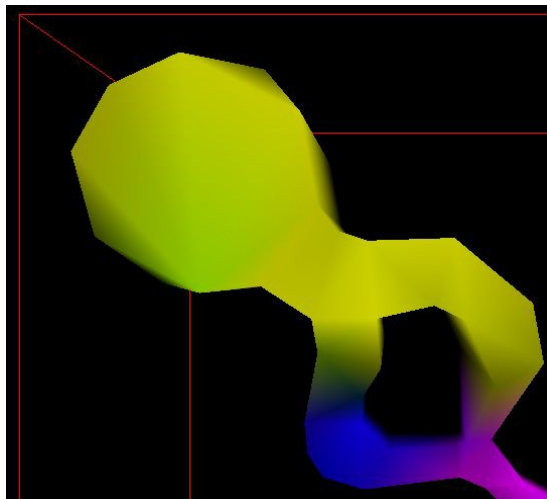
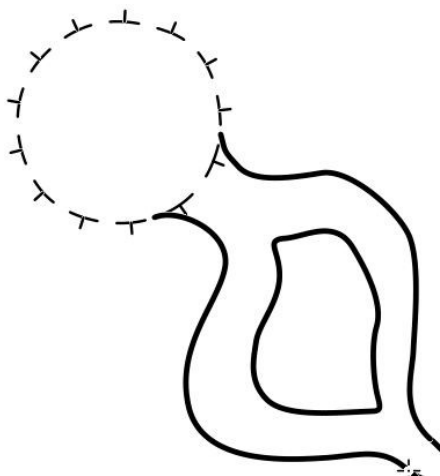


Notez également que la coche jaune est dirigée vers l'extérieur de la cavité car nous voulons que les marques de coche sur la ligne indiquent que le plafond est plus bas dans la galerie qu'il ne l'est dans la cheminée. Lorsque nous compilons notre projet, nous n'obtenons pas ce que nous voulions dans le PDF (ou le modèle *lox*).



Vous pouvez voir que la ligne de bordure du plafond apparaît en travers de la galerie, mais pas au-delà de l'extrémité des parois de celle-ci. Le modèle *lox* montre la même chose. Nous n'avons pas utilisé de lignes de paroi, donc *Therion* a coupé les lignes de symbole rendues pour ne montrer que les parties à l'intérieur de la cavité dont il a déterminé le contour. Si nous voulons simplement que *Therion* montre les symboles en

dehors du contour de la grotte, nous pouvons désactiver le clipping (découpage) pour l'objet. Dans ce cas, nous ferons un clic droit sur la ligne du plafond, et dans le menu contextuel, nous sélectionnerons 'other options/clip/off'. Mais cela ne fixera pas le contour du modèle, et nous devons faire de même pour tous les symboles dessinés dans l'aven. Dans ce cas, nous voulons vraiment que la ligne de la bordure de plafond soit traitée comme une paroi. Pour qu'elle fasse partie du contour de la cavité, nous réglons les options de ligne sur 'other options/outline/out'. Vous pouvez également taper 'outline out' dans le champ d'options du panneau des lignes pour faire la même chose. Maintenant, nous n'avons pas besoin de définir l'option 'clip off' sur la ligne parce qu'elle est traitée comme une ligne définissant le contour (comme les parois sont automatiquement traitées), donc elle ne sera pas coupée dans le dessin. Plus important encore, elle contribue à définir le contour du modèle, de sorte que tous les symboles dessinés à l'intérieur de cette ligne seront rendus dans le PDF, et le modèle lox l'inclura comme faisant partie des murs.



Notre pilier et notre aven sont maintenant rendus correctement à la fois dans le plan et dans le modèle.

Ceci termine la leçon sur les types de lignes. Dans la prochaine leçon, nous apprendrons à utiliser des points pour dessiner des symboles. Les fichiers d'exemple dans le dossier 'Leçon 4' comprennent les lignes de paroi supplémentaires montrées dans cette leçon pour illustrer les piliers et l'utilisation de lignes non murales pour représenter le contour du modèle. Mais comme elles ne font pas partie de la grotte réelle que nous dessinons dans ce tutoriel, elles doivent être supprimées avant de passer à la leçon suivante.

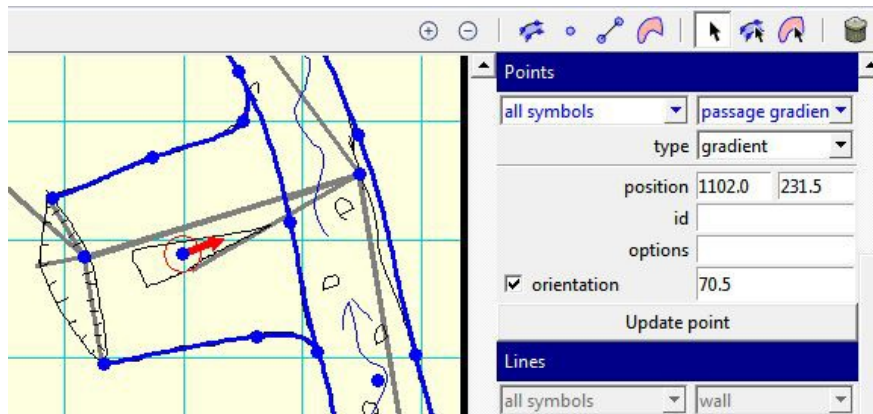
## Leçon 5: Ajouter des symboles et du texte à l'aide de points

Certains symboles sont représentés par des points plutôt que par des lignes. Dans notre section de la galerie de la grotte, nous avons une flèche de pente et quelques symboles d'écoulement de l'eau à dessiner.

Passez en mode "insertion de point" (Ctrl+P) et cliquez au milieu de la flèche de pente sur le croquis *PocketTopo*. Un nouveau point devrait être créé à cet endroit. Comme nous avons créé des points topo la dernière fois, le type de point sera probablement un nouveau point de station topo. Nous pouvons changer le type dans le panneau d'information des points. Le type de point dont nous avons besoin pour une flèche de pente est "gradient". Nous devons également donner une direction à notre flèche (elle doit pointer vers le bas de la pente). Pour ce faire, cochez la case "orientation" dans le panneau d'information. Vous devriez

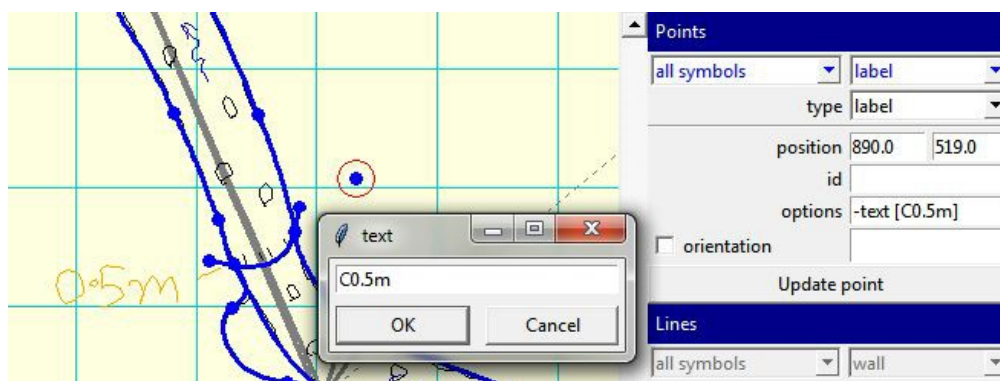
voir qu'une flèche a été dessinée sur votre nouveau point. Vous pouvez maintenant cliquer et faire glisser cette flèche pour la faire pivoter autour du point et l'orienter correctement pour montrer la direction de la pente (la flèche doit pointer vers le bas).

Les flèches de débit d'eau fonctionnent exactement de la même manière que les flèches de pente. Passez



à nouveau en mode "insertion de point" et cliquez sur l'une des flèches d'écoulement d'eau sur le croquis *PocketTopo*. Changez le type de point en "écoulement d'eau". N'oubliez pas de cocher à nouveau la case d'orientation et de définir la direction de la flèche. La flèche d'orientation n'est affichée dans l'éditeur de carte que lorsque le point est sélectionné. Ainsi, si nous ajoutons d'autres symboles, nous devons définir l'orientation de chacun d'entre eux au fur et à mesure, sinon nous devons cliquer sur chacun d'entre eux plus tard pour vérifier leur orientation (ou regarder la carte obtenue pour les vérifier tous). Comme nous sommes toujours en mode "insertion de point", nous pouvons cliquer sur les autres positions où nous voulons des flèches d'écoulement d'eau, en ajustant l'orientation de chacune d'elles au fur et à mesure que nous les ajoutons. Ajoutez quelques flèches d'écoulement de l'eau pour indiquer où le ruisseau coule le long de la galerie.

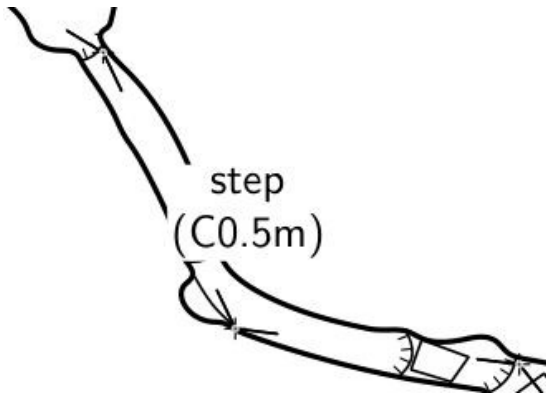
Nous pouvons également utiliser des points pour ajouter des étiquettes de texte à notre dessin. Nous avons plusieurs petites marches dans notre galerie, que nous avons déjà marquées à l'aide du type de ligne "marche au sol" (petit ressaut). Mais il serait bien d'indiquer la hauteur de chaque ressaut. Nous pouvons les étiqueter avec la hauteur de la marche, et indiquer s'il s'agit d'une pente ou d'une montée en utilisant les lettres "P" ou "C". Insérez un nouveau point à côté de la marche marquée 0,5m sur le dessin *PocketTopo*. Changez le type de point en "étiquette". Nous pouvons maintenant faire un clic droit sur le point pour accéder au menu contextuel et sélectionner l'élément "texte". Cela ouvre une zone de texte dans laquelle nous pouvons taper le texte 'C0.5m'.



Notez que cela a pour effet de définir les options du point sur "texte [C0.5m]". Nous pouvons modifier le texte dans le champ des options du panneau des points à droite ou utiliser le menu "clic droit". Nous pouvons également ajouter des options supplémentaires dans le panneau « Points » sur la droite pour contrôler la taille du texte. Si notre texte comprend un espace, le texte dans le champ d'options doit être mis entre guillemets ou entre crochets. Nous pouvons également inclure des sauts de ligne en utilisant le code de caractère <br>, et modifier l'alignement ou le texte en utilisant <center>. Pour illustrer ceci, changez le texte de la boîte d'options en :

```
-text "<center>step<br>(C0.5m) "
```

Le dessin obtenu devrait ressembler à quelque chose comme ceci :



Notre étiquette masque une partie de la galerie de la grotte, nous devons donc déplacer le point un peu plus loin de la galerie dans l'éditeur de carte. Nous pouvons également spécifier une échelle pour le texte afin d'en modifier la taille. Il y a cinq échelles de texte intégrées dans *Therion*. De la plus petite à la plus grande, ce sont les suivantes :

```
xs, s, m, l ,xl
```

Définissez l'échelle en changeant les options pour le point d'étiquette à ceci :

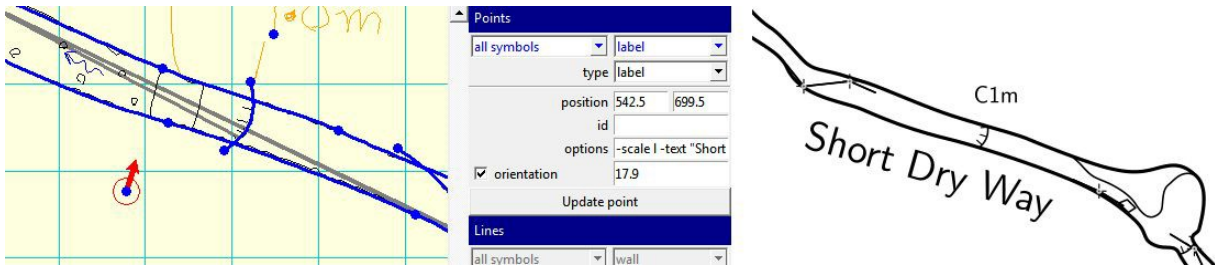
```
-scale m -text "<center>step<br>(C0.5m) "
```

Il s'agit de l'échelle moyenne (m). Ces cinq tailles nous permettent de faire apparaître tous les textes utilisés dans le même but à la même taille. Ainsi, on peut utiliser la grande taille (l) pour les noms de galeries, la moyenne (m) pour les étiquettes de longueurs et de hauteurs, la petite taille (s) pour les étiquettes de fin de galerie (ex. ~30m non topo, etc.). Dans une prochaine leçon, nous verrons comment contrôler la taille des polices utilisées pour chaque échelle de texte. Changez l'échelle en 's' pour cette étiquette, et ajoutez un autre point d'étiquette pour le 'C1m' en amont. Ajoutez un autre point d'étiquetage pour le deuxième 'C1m' et remarquez que le texte du point précédent est automatiquement défini pour le nouveau point. Ainsi, si vous avez plusieurs étiquettes identiques à ajouter à un scrap, vous pouvez définir la première et ajouter chaque étiquette supplémentaire en un seul clic par point d'étiquette.

Nous pouvons également faire pivoter le texte. Créez un point d'étiquette le long de la galerie et définissez le champ d'options comme suit :

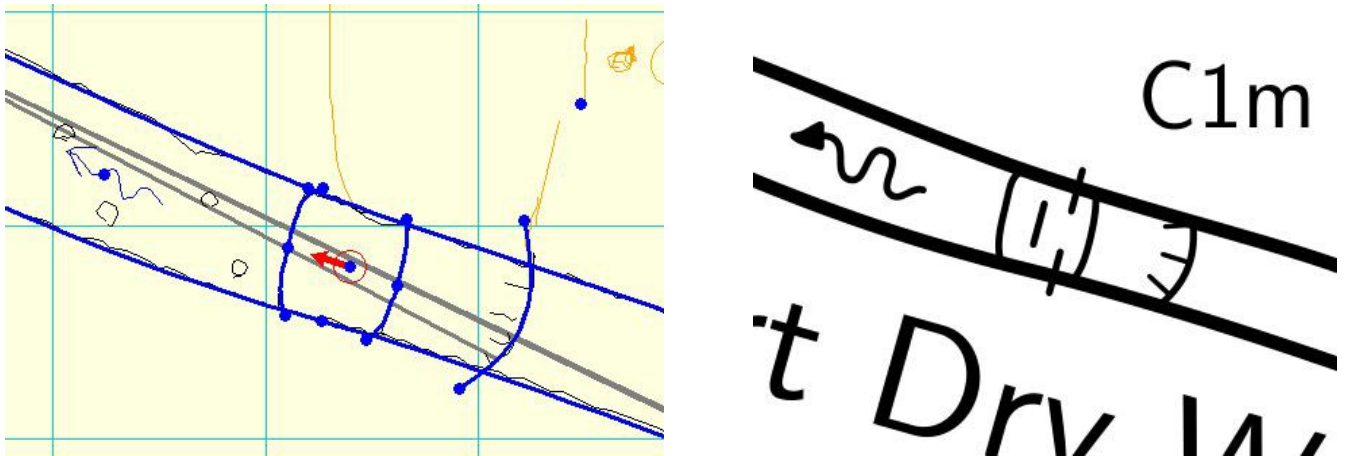
```
-scale 1 -text "Short Dry Way"
```

Le texte traverse le cheminement (cachant une partie de celui-ci) lorsque nous générons le PDF. Nous pouvons faire pivoter le texte de la même manière que nous avons modifié l'orientation des points d'écoulement de l'eau. Cochez la case "orientation" dans le panneau "Points" et ajustez la flèche qui apparaît sur le point de l'étiquette pour qu'elle pointe vers la galerie. Maintenant, le texte est tourné sur le dessin lorsque nous générons le PDF.



Nous pouvons également créer des lignes de type 'label'. Celles-ci peuvent être utilisées pour afficher du texte en utilisant les mêmes chaînes d'options que les points d'étiquette. La différence est que le texte sera étiré pour s'adapter à la longueur de la ligne, et si la ligne est une courbe, le texte sera disposé le long de la courbe de cette ligne. Le problème de la mise à l'échelle du texte pour l'adapter à la longueur d'une ligne est qu'il ne changera pas de taille de la même manière que le texte défini à l'aide de points d'étiquetage. Ainsi, lorsque nous apprendrons comment modifier l'échelle d'une topographie dans une leçon ultérieure, nous constaterons que nos étiquettes de ligne peuvent devenir trop petites pour être lues et qu'elles ne seront pas de la même taille que les étiquettes de points utilisant la même valeur d'échelle. Mais si vous voulez que le texte soit disposé le long d'une courbe, les lignes d'étiquettes (label) sont le moyen d'y parvenir.

Nous allons dessiner un autre élément dans notre galerie pour terminer cette leçon. Juste en dessous de la marche de 1m, il y a un pont de calcite qui a été étiqueté dans le croquis de *PocketTopo*. Nous pouvons le représenter en utilisant une combinaison de lignes et de symboles comme suit. Dessinez deux lignes de type "bordure" pour marquer les bords du pont à l'endroit où ils traversent la galerie. Notez que ces lignes ont été dessinées pour commencer et finir légèrement à l'extérieur des parois de la galerie. Ajoutez ensuite un point symbole sur le pont, en réglant l'orientation pour qu'il pointe vers le bas de la galerie, et le type sur 'flowstone'.



Nous avons également ajouté un autre point d'écoulement de l'eau dans la galerie sous le pont. Nous pouvons voir le rendu de notre pont avec le symbole de la calcite ci-dessus. Le symbole est un peu plus grand que la largeur de notre galerie, il s'étend donc à l'extérieur des parois. Nous apprendrons à corriger cela dans une prochaine leçon. Mais notez que les lignes de bordure que nous avons dessinées pour commencer et finir à l'extérieur des parois sont coupées par les parois dans le résultat obtenu.

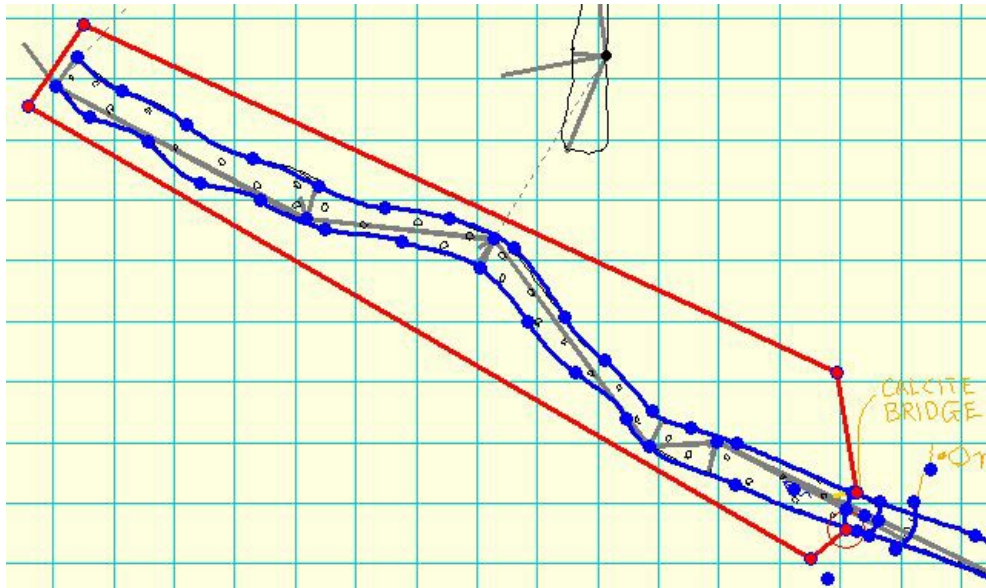
Notre carte commence à ressembler à une véritable étude topographique de cavité. Il existe de nombreux types de points et de lignes à explorer dans *Therion*. Nous pourrions en ajouter d'autres à notre relevé plus tard. Mais parfois, il n'est pas pratique de dessiner individuellement chaque pierre ou grain de sable. Dans la prochaine leçon, nous apprendrons à créer des zones et à utiliser des remplissages de symboles.

## Leçon 6: Remplissage de zones

Certains symboles topo remplissent une zone plutôt que d'apparaître sur un seul point. Dans cette leçon, nous allons créer quelques zones avec des symboles de remplissage. Les symboles de zone typiques montrent les détails de ce qui se trouve sur le sol.

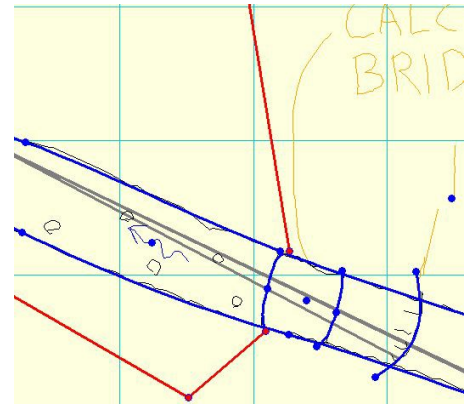
Par exemple, de l'eau, de la boue, du sable, des débris et des rochers. L'extrémité aval de notre galerie a des cailloux sur le sol, donc plutôt que de les dessiner tous, nous allons définir la zone que nous voulons qu'ils recouvrent et ensuite attribuer un symbole pour remplir cette zone. Une aire consiste en une série de lignes définissant la limite d'une zone fermée. Il peut s'agir de lignes de différents types qui apparaissent sur le relevé, ou de lignes invisibles définissant le périmètre de la zone sans apparaître sur le dessin fini. Comme d'autres symboles, le remplissage d'une zone sera délimité par les parois de la cavité dans le scrap. Vous pouvez donc simplement dessiner une zone au-dessus de la galerie, en passant à l'extérieur des parois. Lorsque le dessin sera rendu, seules les parties de la zone situées à l'intérieur des parois seront remplies par le symbole de remplissage de zone.

Nous allons dessiner une zone couvrant toute la galerie en aval du pont de calcite. La ligne de bordure du bord inférieur de notre pont de calcite peut agir comme l'une des lignes définissant notre zone, il nous suffit donc de dessiner une deuxième ligne de bordure partant d'une extrémité de la ligne du pont de calcite, entourant tout la galerie, et se terminant à l'autre extrémité de notre ligne de pont de calcite.

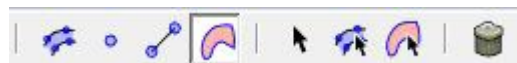




Vous pouvez voir le début et la fin de cette nouvelle ligne dans le détail ici. Cette nouvelle ligne est de type "border" et nous n'avons pas besoin de la rendre invisible car elle se trouve complètement à l'extérieur des parois et donc ne sera pas restituée dans l'étude finale. Remarquez qu'elle est dessinée en longues sections droites plutôt qu'en courbes. Cela n'a pas d'importance pour *Therion* car la ligne doit seulement entourer la galerie entière où nous voulons créer le remplissage du symbole. Mais je trouve qu'il est plus facile de distinguer les lignes de délimitation de ma zone des lignes de détail de la topographie de la cavité si je les dessine en sections rectilignes anguleuses vers les points de la ligne de cheminement, comme ceci.



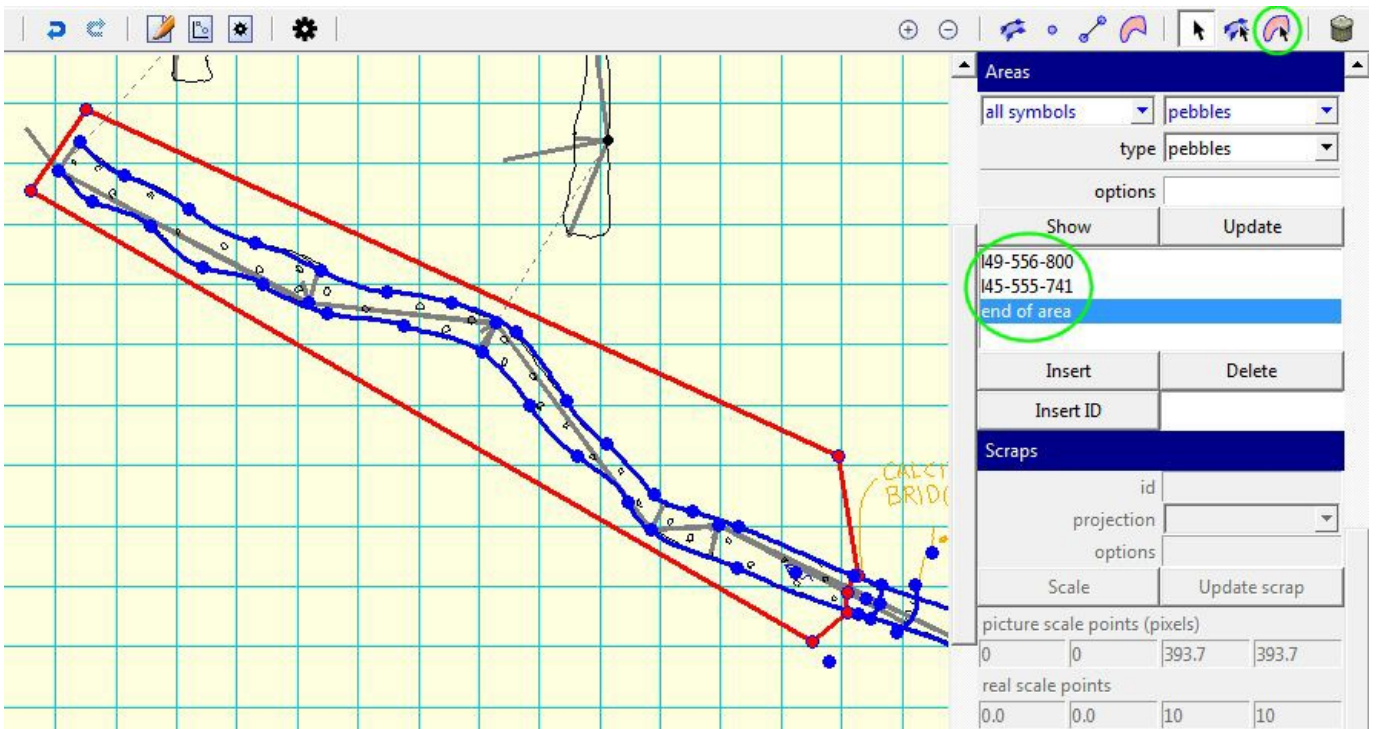
Nous devons maintenant utiliser ces lignes pour définir une zone. Cliquez sur le bouton "nouvelle zone" de la barre d'outils (ou appuyez sur Ctrl+A).



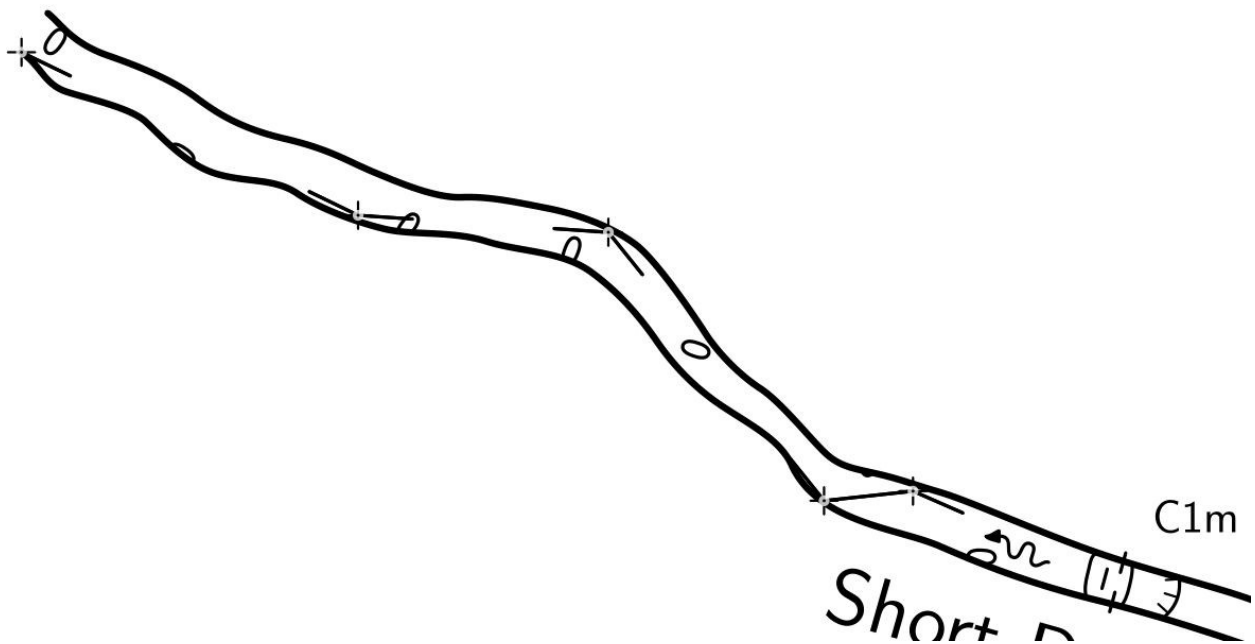
L'éditeur de carte doit maintenant être en mode "insertion de bord de zone", comme l'indique le texte dans la barre d'état rouge. Nous définissons la zone en cliquant sur chaque ligne qui entoure la zone à tour de rôle, en travaillant autour de la zone dans le sens des aiguilles d'une montre ou dans le sens inverse. Dans notre cas, nous n'avons que deux lignes sur lesquelles cliquer. La nouvelle ligne que nous venons d'ajouter, et le bord du pont de calcite sur lequel nous l'avons jointe. Cliquez sur chaque ligne à tour de rôle (l'ordre n'a pas d'importance puisqu'il n'y a que deux lignes). Vous devriez voir l'ID de la ligne ajoutée au panneau d'information "Zones" (identifiée par des codes d'identification longs dans une zone de texte, avec "fin de zone" à la fin de la liste). Lorsque les lignes ont été ajoutées à la zone, appuyez sur la touche "Esc" pour quitter le mode "insertion".

Si vous regardez dans le panneau d'information "Areas", vous devriez voir le type de remplissage de la zone. Par défaut, il est probablement réglé sur "eau". Générez le PDF de la topographie et voyez si votre galerie est maintenant représentée remplie d'eau ? Si ce n'est pas le cas, cela est probablement dû au fait que les lignes définissant la frontière de la zone ont des coches jaunes pointant dans des directions différentes. Elles doivent toutes pointer vers l'intérieur ou l'extérieur de la zone. Sélectionnez l'une des lignes et cliquez sur la case à cocher "reverse" pour inverser la coche de ligne. Maintenant, affichez à nouveau le PDF de la topographie et vous devriez voir une galerie remplie d'eau. Il n'est pas important de savoir de quel côté des lignes se trouve la marque de coche jaune pour les lignes de frontière. Mais si vous créez une zone en utilisant des types de lignes tels que "Floor Step", vous devez définir le côté de la ligne afin que la ligne soit rendue dans le bon sens. Assurez-vous simplement que toutes les lignes définissant la zone sont orientées de la même façon (ticks à l'intérieur ou à l'extérieur de la zone) et le remplissage de la zone devrait avoir un rendu correct.

Deux choses à noter. Si vous sélectionnez à nouveau l'une des lignes, vous constaterez qu'elle possède désormais une valeur "id" dans le panneau d'information "Lignes". Cet identifiant a été généré automatiquement lorsque vous avez ajouté la ligne à la zone. Il est utilisé par la définition de la zone pour identifier les lignes qui la composent. La zone elle-même ne peut pas être sélectionnée dans la fenêtre de dessin. Vous devez donc trouver vos zones en cliquant sur le bouton "Select Next Area" dans la barre d'outils (entouré en vert dans l'image ci-dessous).



Après avoir cliqué sur le bouton "Sélectionner la zone suivante", vous verrez que toutes les lignes définissant votre zone sont colorées en rouge. Si votre esquisse comporte plus d'une zone, cliquez sur ce bouton à plusieurs reprises pour faire défiler toutes les zones une par une. Le panneau d'information sur les zones vous montre également la liste des identifiants de ligne qui définissent votre zone (entourés en vert ci-dessus), ainsi que le type de remplissage attribué à la zone. Changez le type de remplissage en "cailloux" et refaites le rendu du PDF. Vous devriez maintenant voir la galerie remplie de symboles de cailloux. Mais ils sont un peu trop grands pour qu'on puisse en insérer beaucoup. Voici à quoi ressemble le relevé maintenant.



Nous avons fait le tour de ce que nous pouvions montrer avec les remplissages de surface dans notre croquis actuel, donc afin de démontrer d'autres aspects des remplissages de surface, nous allons passer à un nouveau croquis provenant d'une autre séance de topographie dans la cavité. Dans la prochaine leçon, nous verrons comment organiser nos données dans le cadre d'un projet topo plus important, où les données proviennent de plusieurs sorties différentes, et nous combinerons ces croquis dans une étude plus vaste.

## Création d'un second croquis

Fermez le croquis dans l'éditeur de carte *Therion* et suivez le processus décrit dans les leçons 2 et 3 pour générer un nouveau fichier .th et .th2 en utilisant le fichier d'exportation *PocketTopo Therion 'EntToWaterChamberViaPrettyWayth.txt'* (vous pouvez le trouver dans le dossier *PocketTopo* de cette leçon). Nous allons juste dessiner la zone nommée 'Old Grotto'. Nommez donc les fichiers 'OldGrotto.th', et 'OldGrotto.th2'. Il s'agit d'une bonne mise à jour de ce qui a été couvert dans les leçons 2 et 3 avec un ensemble de données différent. Ainsi, plutôt que de répéter toutes les informations de ces leçons ici, essayez de suivre ces leçons et de les appliquer à ces nouvelles données. Cet ensemble de données présente de nouveaux défis car les données du monde réel ne sont pas toujours aussi pratiques que les données du tutoriel que nous avons utilisées jusqu'à présent. Voici donc quelques conseils supplémentaires à appliquer lors de la création de ces nouveaux fichiers.

Tout d'abord, ne nommez pas la topographie "swildons", car nous ne dessinons plus la grotte entière dans notre projet. Nous devons donner à chaque partie de l'étude un nom unique dans le projet. Nommez donc ce relevé 'OldGrotto'. Deuxièmement, ces données exportées par *PocketTopo* contiennent plusieurs séances topo. Ainsi, lorsque vous importez les données de ligne centrale de cheminement, vous obtenez plusieurs blocs de données de ligne centrale. Comme ceux-ci représentent différentes sorties à des dates différentes, nous ne pouvons pas simplement les combiner en un seul bloc de lignes centrales. Nous n'avons besoin que d'un seul bloc de lignes centrales pour la partie de la grotte que nous allons dessiner, mais dans ce fichier de données, la cavité a en fait été topographiée au cours de deux séances. Pour garder les choses simples, nous allons combiner ces séances en un seul bloc (les deux sorties n'étaient séparés que d'un jour) et garder la date du premier jour.

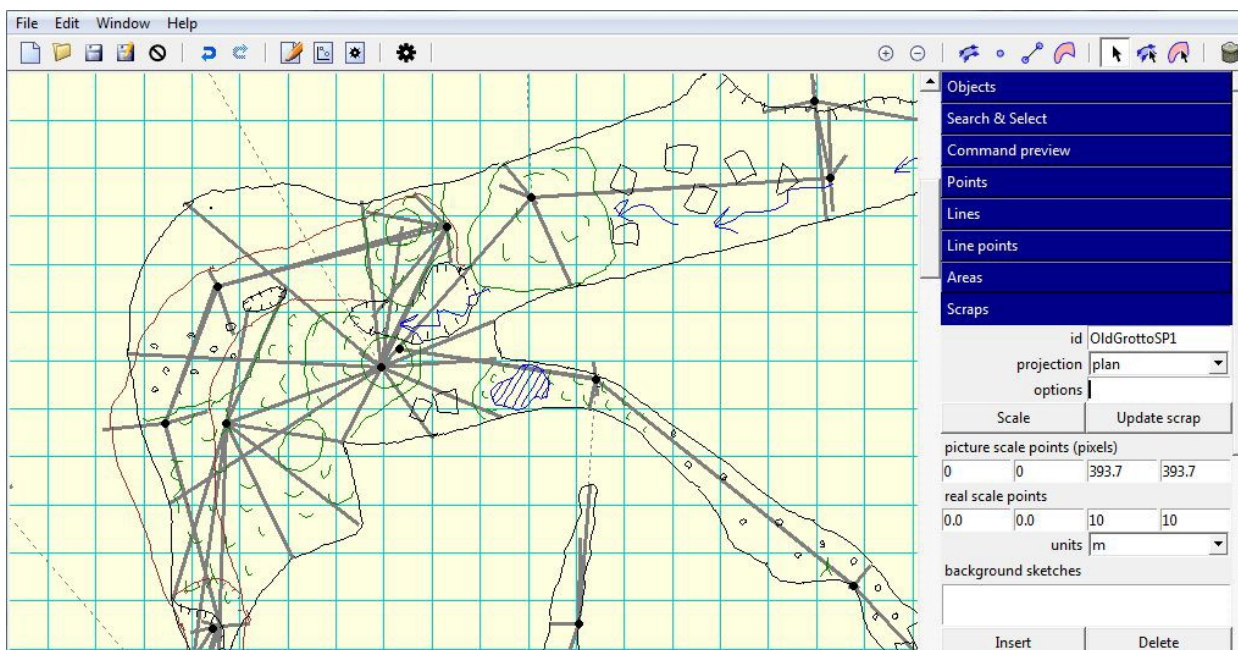
Dans une prochaine leçon, nous verrons comment traiter les zones de dessin couvrant plusieurs séances et que nous ne pouvons pas combiner en un seul bloc de ligne centrale. Pour l'instant, supprimez l'ensemble du premier bloc de ligne centrale et déplacez toutes les données des segments topo du dernier bloc (avec la date 2013.02.24) dans le bloc avec la date 2013.02.23. Nous devons également ajouter un lien manquant entre deux des séries qui est absent des données du relevé topo. Ainsi, après la ligne "date" dans le bloc de la ligne centrale, ajoutez la ligne :

```
equate 4.3 6.0
```

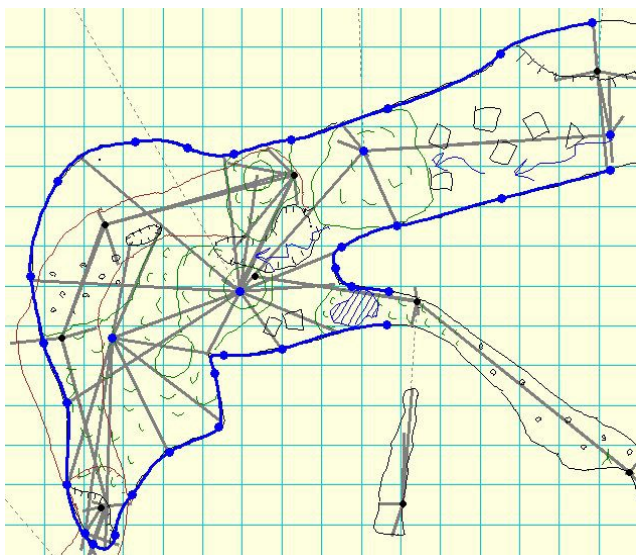
Vous devrez changer la ligne 'source' dans votre fichier *thconfig* pour traiter 'OldGrotto.th' maintenant, afin de pouvoir vérifier que *Therion* est capable de traiter votre fichier correctement. Mettez également en commentaire les exportations pour le format kml, car tant que nous n'aurons pas lié notre nouvelle topographie à un emplacement géographique (en utilisant un repère d'entrée), *Therion* enverra des avertissements indiquant qu'il ne peut pas générer de sortie kml. Il devrait compiler maintenant pour générer un PDF qui montre toutes les données de la ligne centrale de ce relevé topo, puisque nous n'avons pas encore inclus de scraps.

Lorsque vous créez le fichier th2, suivez uniquement la Leçon 3 jusqu'au moment où vous avez créé un scrap et défini l'échelle. Nommez ce scrap 'OldGrottoSP1'. Nous reprendrons le dessin dans cette esquisse à partir de ce point dans cette leçon. Si vous êtes bloqué, vous trouverez des copies des fichiers que vous devriez avoir à ce stade, dans un dossier " templates " situé dans le dossier de cette leçon.

Le croquis d'arrière-plan que nous avons importé couvre une grande partie de la grotte. Vous allez devoir faire un zoom avant et faire défiler la carte pour montrer la zone que nous allons dessiner pour le reste de cette leçon. La salle Old Grotto devrait apparaître comme suit dans votre éditeur de carte :



Nous voulons seulement ajouter des points de station pour les stations dans la salle de la Vieille Grotte, car nous allons seulement dessiner cette partie de la cavité dans cette leçon. Cliquez donc uniquement sur les points de station indiqués en bleu ci-dessous, et ajoutez les lignes de paroi comme indiqué.



Nous n'avons ajouté que des points de station pour la ligne de cheminement traversant le niveau supérieur de la salle jusqu'à la station 2.61 (qui se trouve en bas à droite et semble être un point sur la paroi dans cette capture d'écran). La série de visées qui passent sous la salle dans le chenal d'écoulement doit être dessinée dans un autre format, que nous aborderons dans une leçon ultérieure. Notez comment nous avons arrêté le dessin des parois latérales de la galerie à des points face à face les uns aux autres, et légèrement dans la galerie latérale. Cela nous aidera à joindre nos croquis plus tard.

Nous devons maintenant informer *Therion* de notre nouveau scrap. Dans le fichier .th situé au-dessus du bloc de la ligne centrale de cheminement, nous devons donc ajouter la référence au fichier .th2 et définir une carte contenant notre scrap :

```
survey OldGrotto input
OldGrotto.th2 map
OldGrottoMP
OldGrottoSP1
endmap
centreline
.....
```

Maintenant, lorsque vous compilez le projet, vous devriez obtenir un PDF contenant ce scrap de la cavité au lieu de la ligne de cheminement du relevé topo. Nous pouvons maintenant nous remettre à dessiner d'autres zones de remplissage.

En commençant par la zone de coulée de calcite entourant la station 2.58, nous devons réfléchir aux lignes que nous utiliserons pour définir les limites de cette zone. Nous pouvons dessiner une ligne de bordure pour les bords de cette coulée stalagmitique qui sont à l'intérieur de la galerie. Mais cela ne complète pas la zone entière. Nous aurions pu continuer cette frontière pour fermer l'espace et faire une zone complète, mais nous allons utiliser une partie de cette frontière pour définir la zone de rochers à droite aussi. Au lieu de cela, nous voulons la diviser en sections. Si nous ajoutons une deuxième section de ligne de bordure (à l'extérieur des parois pour qu'elle n'apparaisse pas dans le croquis), nous pouvons ajouter la zone, cliquer sur ces deux segments de ligne pour les ajouter à la zone, définir le type de remplissage sur "flowstone" et compiler le projet pour voir notre zone remplie.

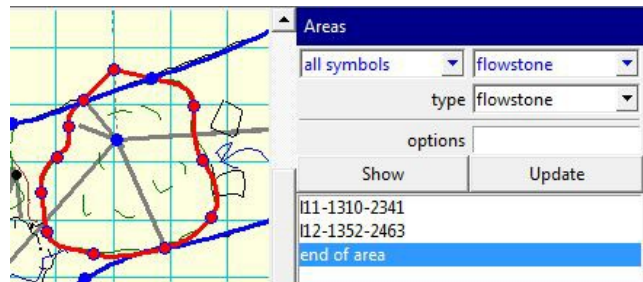
Ligne de bordure 1



Ligne de bordure 2



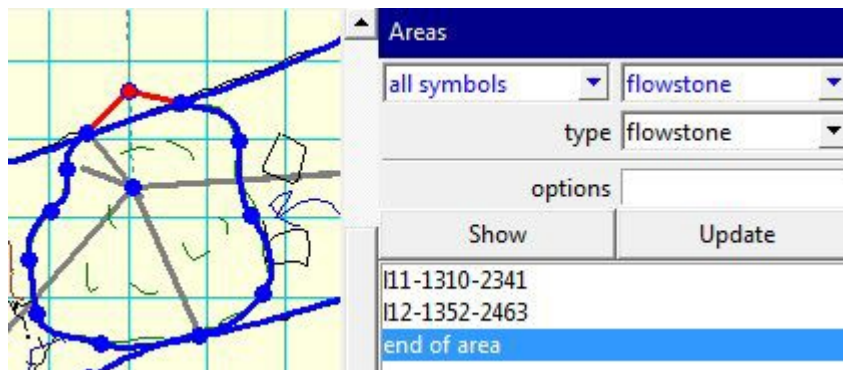
Zone complète comprenant les deux lignes



Les problèmes commencent lorsque nous voulons utiliser une partie de la ligne entourant la coulée stalagmitique pour définir une deuxième zone pour les rochers à sa droite. Si nous divisons cette ligne afin de pouvoir utiliser uniquement la moitié droite de celle-ci pour notre zone adjacente, nous brisons notre zone précédente. En effet, lorsque nous divisons la ligne, l'ID de la ligne ne peut pas être attribué aux deux fragments de la ligne. *Therion* supprime donc l'identifiant de la ligne, et nous avons maintenant une zone que nous avons définie pour la coulée stalagmitique en utilisant un identifiant de ligne qui n'est plus présent dans l'esquisse. Si vous essayez de compiler le projet après avoir divisé cette ligne, *Therion* signalera une erreur 'l'objet n'existe pas I1113102341'. L'objet manquant 'I1113102341' est l'identifiant de la ligne dans notre zone qui n'existe plus. Vous pouvez voir cet identifiant dans le panneau d'information de la zone ci-dessus. Nous devons donc sélectionner à nouveau la zone (notez que seule une partie de la zone est colorée en rouge, car c'est la seule ligne dont l'ID correspond à la zone).



La ligne de séparation



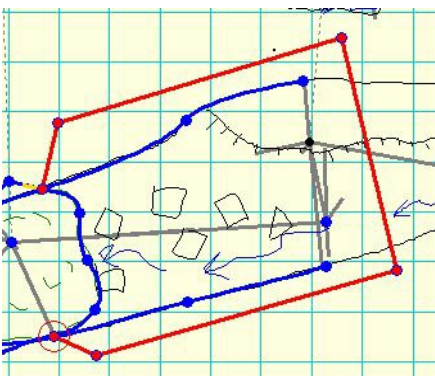
La zone sélectionnée après division, avec une seule ligne en surbrillance.

Cliquez sur l'identifiant dans la liste des identifiants de zone, puis cliquez sur le bouton "supprimer" sous la liste pour le supprimer. Maintenant, cliquez sur le bouton "insérer" sous la liste, et cliquez sur les deux moitiés de la ligne entourant la coulée à tour de rôle. Cela va attribuer à ces lignes de nouveaux ID et les ajouter à la définition de la zone.

Enfin, appuyez sur la touche "Esc" pour sortir du mode "Insertion de ligne" pour la zone. Le projet devrait compiler à nouveau et afficher le remplissage de la zone de coulée de calcite dans la correction du PDF. Notez que les ID réels qui seront attribués à vos lignes seront différents de ceux montrés dans ce tutoriel, car *Therion* génère les ID automatiquement lorsqu'ils sont requis.

Ceci devrait, je l'espère, vous avoir expliqué pourquoi vous devez être très prudent lorsque vous utilisez des lignes qui font partie de votre esquisse visible pour définir des zones. C'est parfois la seule façon d'obtenir le dessin que vous souhaitez. Mais dans la mesure du possible, évitez d'utiliser des lignes qui ont une autre fonction dans l'esquisse pour les bordures de vos zones (comme les parois) et dessinez plutôt des lignes de bordures invisibles distinctes pour définir les zones de remplissage. Si vous devez utiliser des lignes existantes, réfléchissez bien à la façon dont vous voulez les diviser et essayez de les diviser avant de les utiliser pour les zones. Vous éviterez ainsi de devoir corriger les zones par la suite.

Nous pouvons maintenant définir une deuxième zone à droite de la coulée stalagmitique pour la remplir de rochers. Nous allons utiliser la moitié droite de la ligne de bordure de la coulée comme une partie de la limite de la zone. Dessinez donc une autre ligne de bordure entourant le reste de la zone, en commençant à une extrémité de la ligne de bordure de coulée, et en terminant à l'autre extrémité de cette ligne.

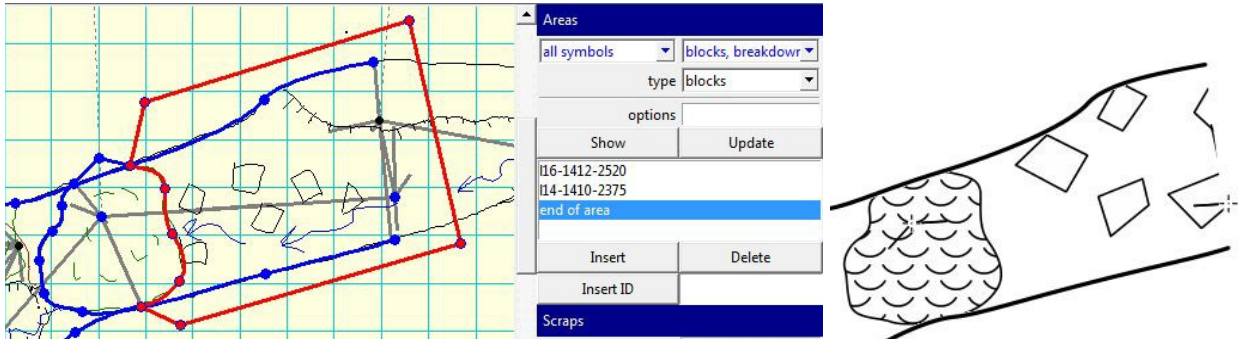


Comme précédemment, nous l'avons dessinée à l'aide de sections de lignes droites afin de la faire ressortir visuellement et de savoir qu'il s'agit d'une bordure de zone. Nous n'avons pas besoin de la rendre invisible, car cette ligne se trouve entièrement à l'extérieur des parois de la galerie et ne sera donc pas restituée dans l'esquisse.

Créez maintenant une nouvelle zone, cliquez sur cette nouvelle ligne, puis sur la ligne qui borde la zone de la coulée. Appuyez sur 'Esc' pour quitter le mode 'insertion de bordure de zone', puis changez le type de zone en 'blocs'. Compilez le PDF et vérifiez que les blocs apparaissent. Si ce n'est pas le cas, il est probable que vos deux lignes de bordure définissant cette zone n'ont pas leurs tics jaunes pointés dans la même direction. Si c'est le cas, vous devez "inverser" l'une des lignes de bordure de sorte que les deux lignes aient leurs tics pointés vers la zone ou

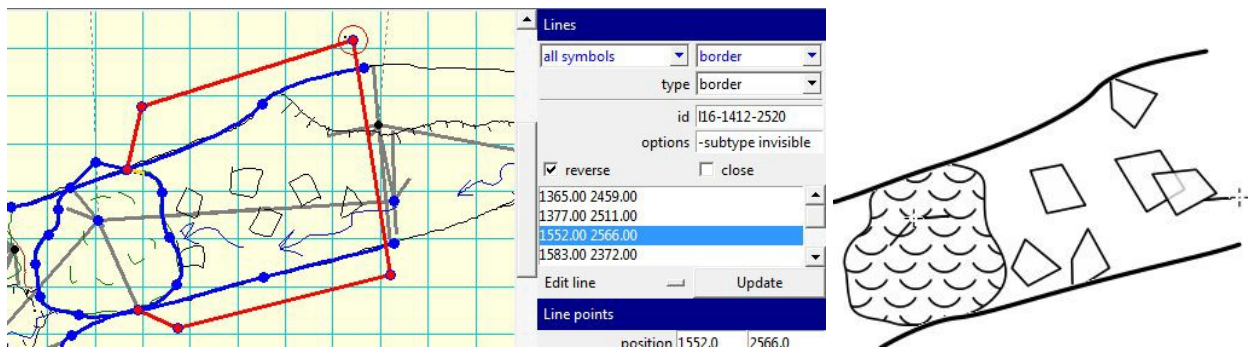
vers l'extérieur. Faites-le avec la nouvelle ligne de bordure que nous venons d'ajouter, plutôt qu'avec la ligne que nous partageons avec la zone de coulée stalagmitique. Sinon, nous briserons la zone la coulée lorsque nous calculerons cette ligne.

Maintenant, nos deux zones de remplissage apparaissent dans notre PDF. Mais remarquez comment les rochers sont coupés en deux sur le bord droit de la zone de remplissage des blocs. Ceci est dû au fait que *Therion* adaptera le motif de remplissage à l'intérieur des limites de la zone, mais notre zone s'étend à



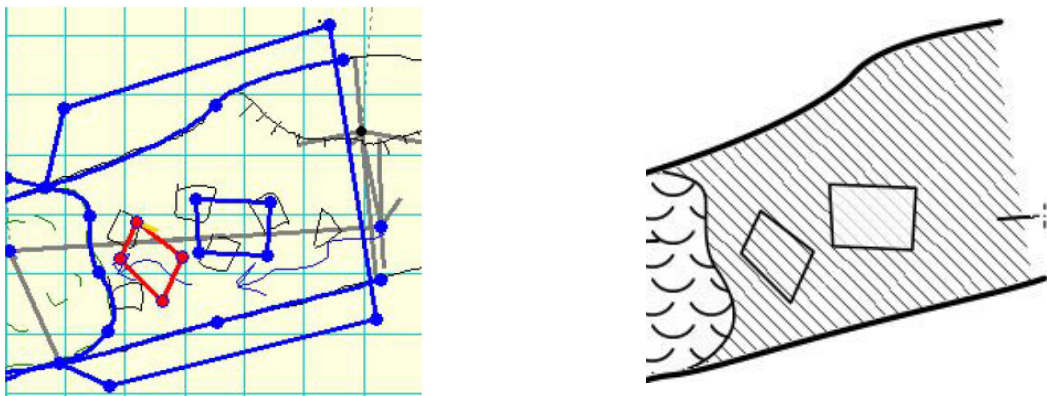
l'extérieur des parois et le dessin de certains blocs est coupé par ces parois. L'aspect n'est pas si mauvais là où les rochers sont coupés par les parois, mais la fin de notre fragment a quelques 'demi rochers'. Nous pouvons résoudre ce problème en déplaçant la ligne de bordure sur cette extrémité ouverte du scrap de manière à ce qu'elle soit juste à l'intérieur de la galerie. Mais maintenant qu'elle est à l'intérieur de la galerie, elle sera dessinée comme une ligne de bordure solide, nous devons donc rendre cette ligne invisible également.

Faites glisser les points de contrôle sur la ligne de manière à ce qu'elle se trouve juste à l'intérieur des extrémités des parois de la galerie, puis cliquez avec le bouton droit de la souris sur la ligne et changez le sous-type en "invisible". Maintenant, nous pouvons voir que le PDF ne dessine pas de demi-blocs à travers l'extrémité ouverte de la galerie.

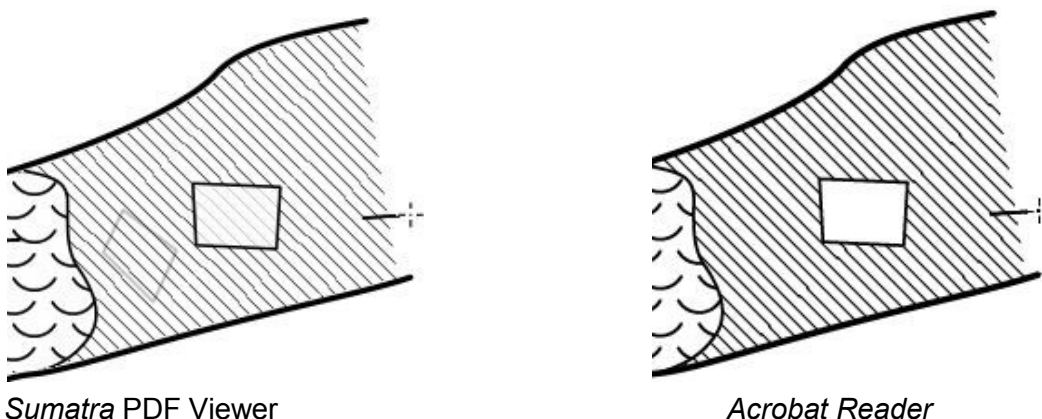


Les blocs rocheux ont également bougé, car leur emplacement est aléatoire. Vous constaterez qu'ils apparaissent à un endroit différent chaque fois que vous recompiliez le projet. Si vous souhaitez placer un rocher important à un endroit précis, vous devez le dessiner à l'aide de lignes de bordure de rocher, comme indiqué dans la leçon 4.

Les lignes de bordure de roche se comportent différemment des lignes de bordure normales lorsqu'elles sont utilisées avec des remplissages de zone. Pour illustrer cela, changez le type de zone que nous venons de dessiner en "eau". Ensuite, dessinez deux gros rochers dans la zone. Celui de gauche est dessiné avec le type de ligne "border", tandis que celui de droite est dessiné avec le type de ligne "rock border". Lorsque vous compilez le projet, vous pouvez voir que le rocher "bordure rocheuse" apparaît au-dessus de l'eau, alors que celui dessiné avec une simple ligne "bordure" ne cache pas du tout le remplissage d'eau.



Changez le type de ligne du rocher de gauche en "bordure de rocher", afin que les deux rochers soient rendus comme s'ils sortaient de l'eau. Si nous voulions dessiner un rocher qui se trouve sous la surface de l'eau, nous pouvons le faire en utilisant l'option " place ". Cliquez avec le bouton droit de la souris sur la ligne de bordure du rocher de gauche et, dans le menu contextuel, sélectionnez "Autres options > Placer > Fond". Ou bien, tapez simplement 'place bottom' dans le champ d'options de cette ligne. Maintenant, lorsque nous compilerons le projet, ce rocher sera rendu sous le motif de remplissage de la zone d'eau. Notez que tous les visionneurs de PDF ne prennent pas en charge la transparence dans leur rendu de PDF. Comme vous pouvez le voir ci-dessous dans l'image de droite, le bloc rocheux sous la zone de remplissage de l'eau est complètement caché dans le visualiseur PDF d'Acrobat Reader, et le bloc rocheux au-dessus de l'eau ne montre pas non plus le remplissage de l'eau en dessous de lui. La visionneuse PDF *Sumatra* rend le PDF par transparence comme on le voit à gauche.

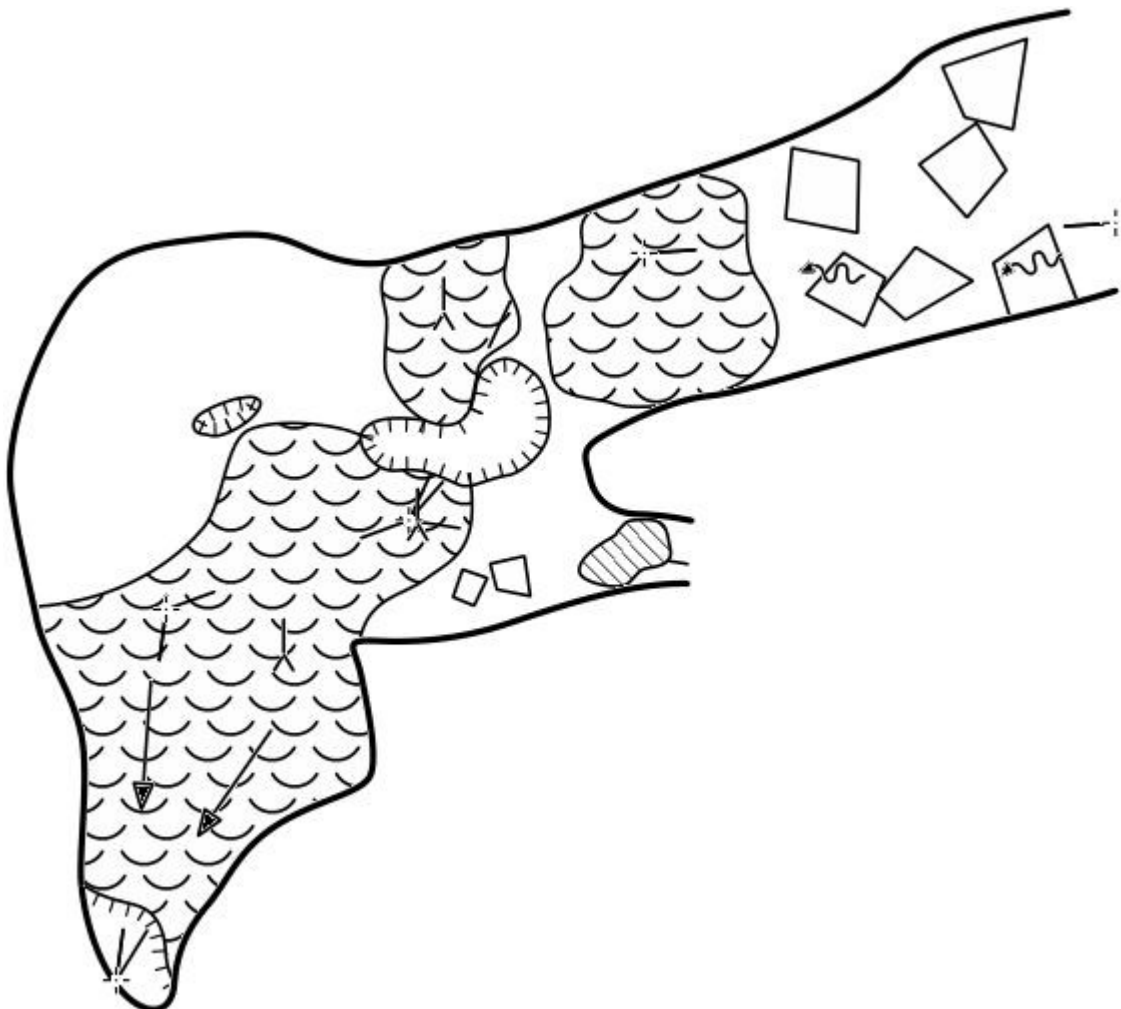


Blocs rocheux sous et au-dessus de l'eau

Vous pouvez trouver une copie du fichier d'esquisse avec ces rochers dans l'eau dans le dossier de cette leçon sous le nom de fichier 'OldGrottowaterfill.th2'. Mais pour terminer cette leçon, nous allons supprimer



ces rochers et changer le remplissage de la zone en " blocs " afin qu'il reflète correctement la cavité que nous dessinons. Entraînez-vous à dessiner les autres zones de la salle de l'Old Grotto, sans oublier les orifices dans le sol (utilisez le type de ligne 'pit' pour ceux-ci). La version finale 'OldGrotto.th2' de cette leçon comporte également des points d'écoulement d'eau, de stalagmite et de gradient. Le croquis devrait ressembler à ceci lorsque vous aurez terminé.



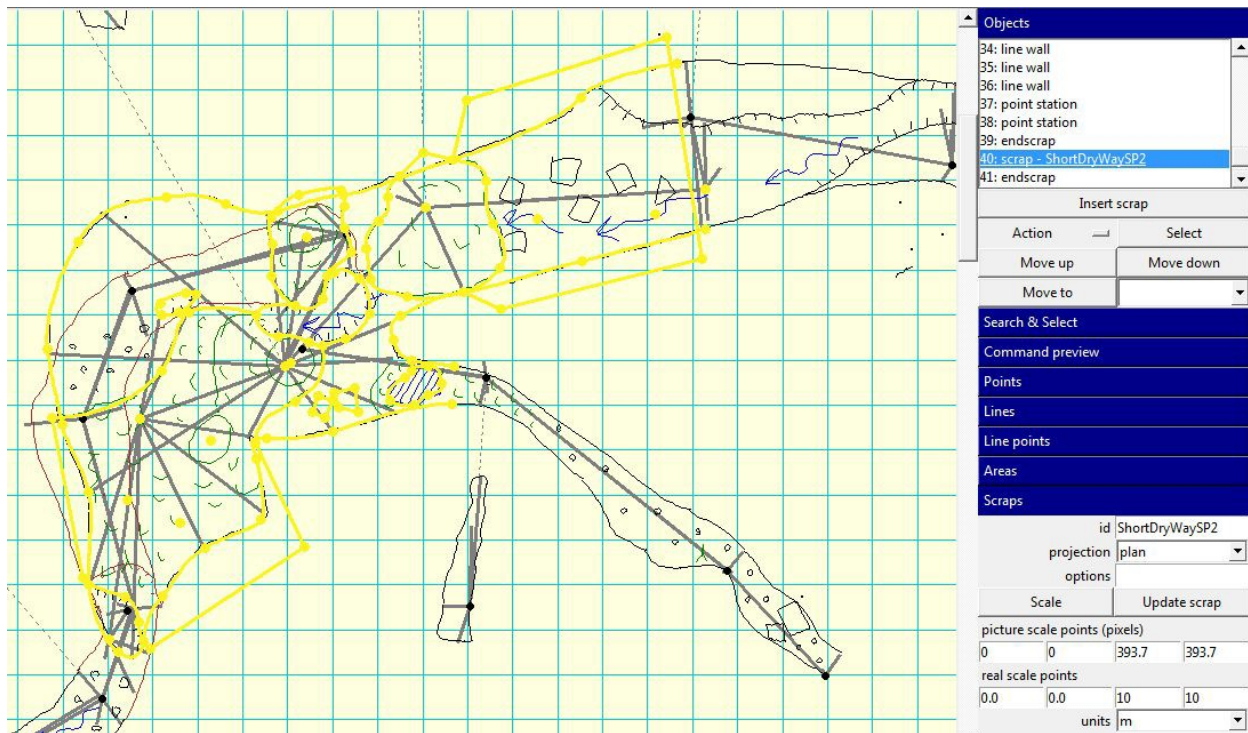
Jusqu'à présent, nous avons ignoré le cheminement du ruisseau qui passe sous la salle parce qu'un scrap n'est pas autorisée à se chevaucher lui-même. Dans la prochaine leçon, nous apprendrons à ajouter d'autres scraps au projet et à les combiner en un plus grand relevé.

## Leçon 7: Travailler avec plusieurs calques (scraps)

Notre grotte possède un ruisseau qui passe sous une partie de la salle de la Vieille Grotte d'où l'eau se perd à travers un trou dans le sol. Nous ne pouvons pas inclure à la fois ce cheminement du ruisseau et la salle dans un seul calque car un scrap ne peut pas se chevaucher lui-même. Dans cette leçon, nous allons donc apprendre à ajouter des calques supplémentaires à notre dessin. Une fois que vous saurez faire cela, vous serez en mesure d'établir des relevés, même pour les plus grands systèmes souterrains.

Il n'y a pas de raison de ne pas dessiner plusieurs morceaux dans le même fichier de dessin, sauf s'il devient difficile de voir ce qui se passe dans l'éditeur. *Therion* rendra le relevé final de la même manière, que les fragments soient dans un ou plusieurs fichiers .th2. Nous allons commencer par dessiner un nouveau scrap dans le même croquis que celui que nous utilisons déjà pour la salle. Pour rester simple, nous allons d'abord dessiner la galerie latérale qui ne chevauche pas la salle, car cela sera plus clair dans le tutoriel.

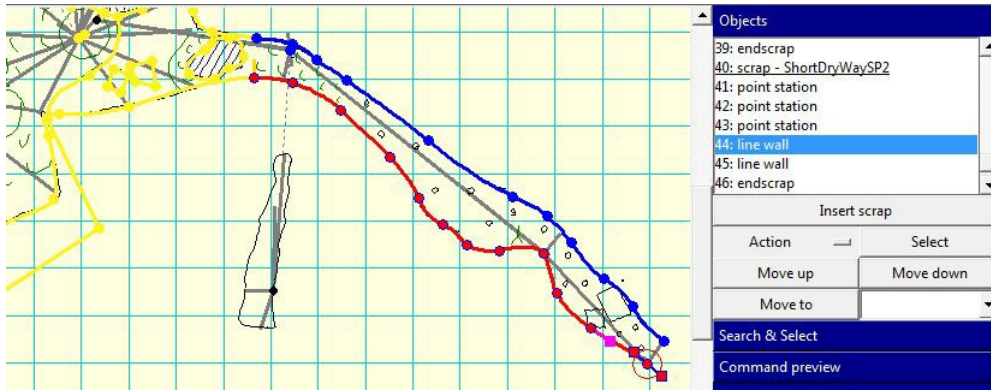
Commencez par insérer un nouveau scrap en cliquant sur le bouton de la barre d'outils (ou en appuyant sur Ctrl+R). Nommez le scrap 'ShortDryWaySP2' et définissez la projection sur 'plan' en suivant les mêmes étapes que celles de la leçon 3.



Si vous regardez le panneau d'information "Objets" en haut du volet de droite, vous devriez voir qu'il y a maintenant deux calques (scraps) définis. Chacun d'eux commence par une ligne "`scrap <name>`" où `<name>` est le nom du scrap, et se termine par la ligne "`end scrap`".

Tous les objets de chaque scrap apparaissent entre ces lignes. Notre nouveau scrap ne contient donc rien pour l'instant. Si vous sélectionnez l'une des lignes de départ d'un scrap, tout ce qui se trouve dans ce calque est coloré en bleu dans l'éditeur de carte.

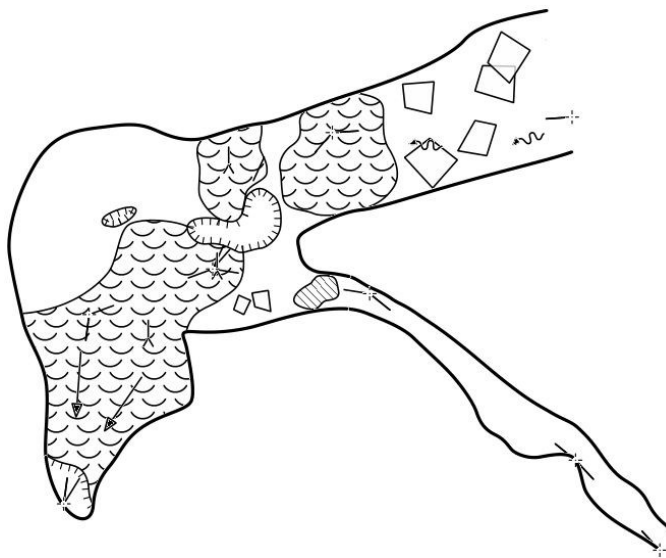
Sélectionnez la ligne "`scrap ShortDryWaySP2`" et tout devrait devenir jaune dans le dessin car nous n'avons encore rien dessiné dans ce scrap. Maintenant que le nouveau scrap est sélectionné dans la fenêtre des objets, toute nouvelle ligne ou tout nouveau point que nous dessinerons sera placé dans le nouveau scrap. Commencez par ajouter les points de la station topo pour la galerie latérale afin que *Therion* sache où placer le scrap dans la topographie. La procédure est exactement la même que celle de la leçon 3. Ensuite, dessinez les lignes des parois et les rochers. N'oubliez pas de vérifier que les tics sur les lignes de paroi pointent vers la galerie. Cela devrait suffire pour ajouter le nouveau calque dans le relevé. Voici à quoi ressemble notre scrap dans l'éditeur de carte.



Notez que dans la liste des objets, notre scrap contient maintenant 3 points de station et deux lignes de paroi. Si nous compilons le projet, nous ne verrons pas encore de différence dans le PDF, car nous n'avons pas ajouté ce nouveau scrap à la carte que nous produisons. Mais vous verrez que les parois ont été dessinées dans le modèle 3D si vous ouvrez le fichier .lox. Pour ajouter ce nouveau scrap à la carte, il suffit d'ajouter le nom du calque dans la déclaration de carte du fichier OldGrotto.th.

```
map
OldGrottoMP
OldGrottoSP1 S
hortDryWaySP2
endmap
```

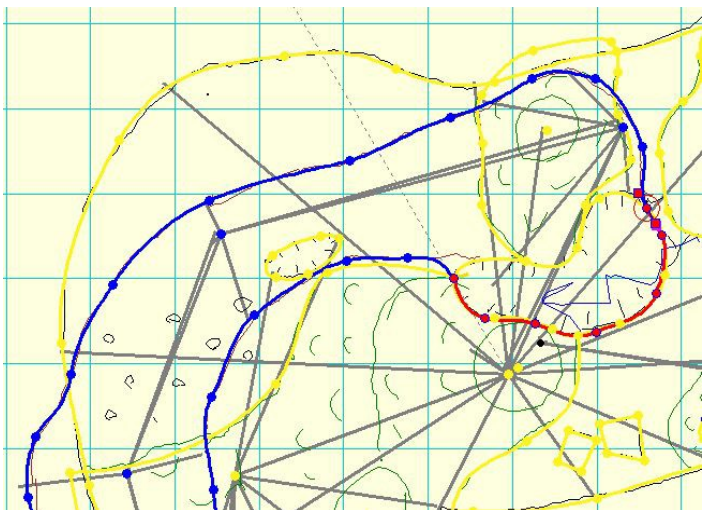
Maintenant, compilez à nouveau le projet pour voir nos deux scraps dans le PDF.



Ensuite, nous allons dessiner la galerie du ruisseau qui passe sous la salle. Nous pouvons le dessiner dans le même fichier d'esquisse (.th2), mais il traversera (ou passera sous) le dessin de la salle. Le fait d'être dans un scrap séparé signifie que le scrap actif sera représenté en bleu, tandis que les autres scraps seront colorés en jaune. Il n'est donc pas trop difficile de voir ce que nous faisons. Créez un nouveau scrap nommé 'OldGrottoStreamSP1', en veillant à définir le type sur 'plan', et ajoutez les points topo dans le cours du ruisseau (6.1 à 6.6). Nous pourrions étendre le cours d'eau au-delà de ce point plus tard.



Voyez les 6 points de station que nous avons ajoutés (les points bleus) dans le nouveau calque ci-dessus. Maintenant nous pouvons dessiner les parois de cette galerie sous-jacente. À l'extrémité amont, nous avons dessiné une extrémité fermée de la galerie autour de la paroi arrière de l'orifice descendant de la salle ci-dessus. Cela donnera un meilleur modèle 3D que de laisser cette extrémité de galerie ouverte.

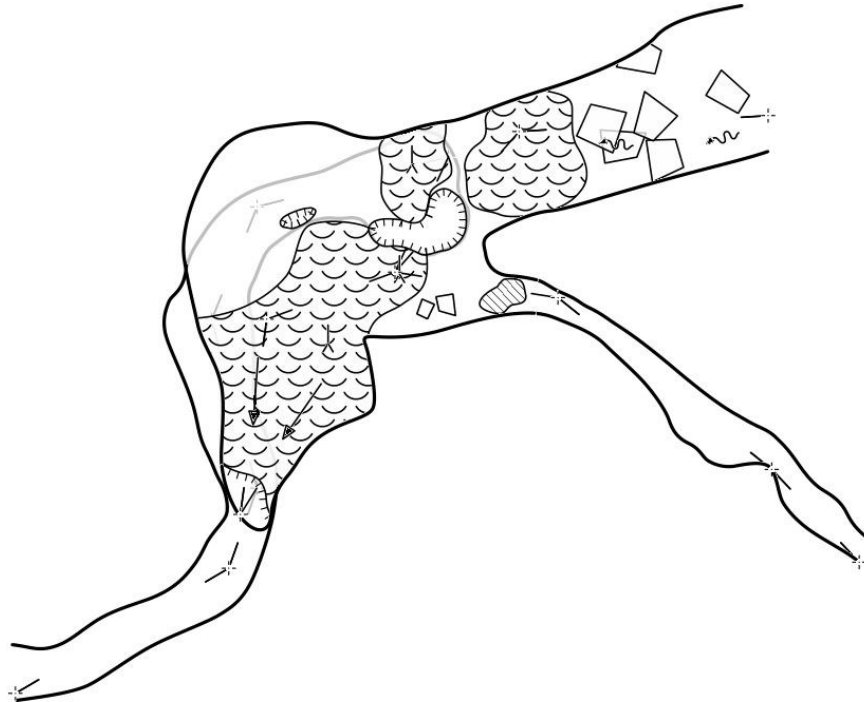


Vous pouvez voir cette ligne de paroi supplémentaire autour de la fin de la galerie surlignée en rouge ici. N'oubliez pas d'ajouter ce nouveau scrap dans la définition de la carte dans le fichier OldGrotto.th. Maintenant, lorsque nous compilons le projet, nous devrions voir le passage du ruisseau dans le PDF. Mais il n'apparaît pas comme étant sous la salle. Nous devons dire à *Therion* de restituer ce nouveau secteur dans une couche séparée en dessous des autres. Pour ce faire, nous ajoutons le mot clé 'break' dans la définition de la carte comme suit :

```
map
  OldGrottoMP
  OldGrottoSP1
  ShortDryWaySP2
```

```
break
OldGrottoStreamSP1
endmap
```

Maintenant, notre PDF devrait ressembler à ceci :



Si vous regardez attentivement, vous pouvez voir qu'il y a un petit espace dans les parois où notre scrap de galerie latérale et le scrap de la salle se rencontrent. Lorsque nous dessinons des scraps dans différentes esquisses, les espaces entre les extrémités des parois sont souvent bien plus visibles. *Therion* peut rapprocher les extrémités des lignes de parois pour joindre les scraps en douceur. Il suffit d'indiquer au programme quels sont les calques que nous voulons joindre correctement. Ceci est fait en ajoutant un 'join' dans le fichier .th qui inclut le fichier d'esquisse contenant les scraps que nous voulons joindre. Ajoutez une ligne juste après la ligne 'input OldGrotto.th2' :

```
join OldGrottoSP1 ShortDryWaySP2
```

Maintenant, compilez à nouveau le projet et l'écart entre ces deux zones aura disparu. Nous n'avons pas besoin de joindre le scrap OldGrottoSP1 au scrap OldGrottoStreamSP1 car ceux-ci n'ont pas de lignes de paroi communes (qui se joignent bout à bout).

## Leçon 8 : Structures de données de projets plus complexes

Les topographies du monde réel sont rarement réalisées en une seule séance. Dans cette leçon, nous allons réorganiser notre projet de manière à pouvoir l'étendre pour couvrir plusieurs sorties topo. Il est important d'avoir une bonne structure de données pour vos fichiers dans les grands projets. Il existe de nombreuses façons de procéder, et les différents utilisateurs de *Therion* ont leur propre façon d'organiser

leurs données. La structure de données que nous allons utiliser pour cette leçon est un exemple de l'une de ces façons de travailler. Elle a l'avantage de garder vos fichiers organisés par séance topo et vous permet de traiter chaque sortie séparément lorsque vous éditez seulement une partie de la cavité. Vous pouvez ensuite traiter l'ensemble de la grotte pour générer la carte et les modèles complets.

Nous allons regrouper nos fichiers de données et de croquis pour chaque sortie topo. Vous pouvez le faire simplement en nommant les fichiers avec le même préfixe. Pour les projets de systèmes souterrains plus importants, je regroupe également les séances en créant un dossier pour chaque zone ou grotte du système (en général, vous commencez par des grottes nommées séparément qui, au fil du temps, sont connectées en un seul système, mais continuent d'être désignées par le nom original de chacun des entrées principales). L'organisation des données est également importante dans notre façon de nommer et de structurer les blocs de relevé du projet.

Le schéma de la structure des données que nous allons utiliser est le suivant :

```
survey EntireCave

  < définitions des cartes pour l'ensemble de la grotte >

survey Trip1
  < croquis d'entrée pour la séance 1>
  < définitions des cartes pour la séance 1>
  centreline
    <données topo séance 1>
  endcentreline
endsurvey #end of Trip1

survey Trip2
  < croquis d'entrée pour la séance 2>
  < définitions des cartes pour la séance 2>
  centreline
    <données topo séance 2>
  endcentreline
endsurvey #end of Trip2

<autres séances éventuelles >

endsurvey #end of EntireCave
```

Nous pourrions entrer cette structure de données dans un grand fichier .th, mais il est généralement plus facile de séparer chaque séance topo dans son propre fichier. Ainsi, chaque sortie a un fichier .th contenant les données de la ligne de cheminement (numéros), plus un ensemble de fichiers .th2 qui l'accompagnent et qui contiennent tous les dessins. Il est à noter que les dessins de scrap (croquis dans les fichiers .th2) sont inclus dans le bloc de données de la topographie pour chaque trajet. C'est important car cela permet à *Therion* de comprendre quelle station numérotée dans les données correspond à quel point de station numéroté dans les scraps. Par exemple, nous pouvons avoir une séance où l'entrée a été étudiée en utilisant les numéros de station 1 à 25. Puis une séance ultérieure où une galerie plus loin dans la grotte a été topographiée en utilisant les stations numérotées 1 à 32. Cela signifie que nous avons deux stations utilisant le numéro 20. Si les scraps sont inclus en dehors du bloc de données de la topographie sur la séance topo individuelle, nous devons nommer chaque station dans le scrap avec le nom de la séance et

le numéro. Mais en incluant le scrap à l'intérieur du bloc de la topographie sur les séances, nous pouvons simplement utiliser le nom '20' dans le dessin du scrap et *Therion* sait à quelle séance il appartient. Ceci est particulièrement utile lorsque l'on dessine des scraps importés de *PocketTopo*, car l'éditeur de cartes de *Therion* définira automatiquement les noms des points de station à partir du croquis de *PocketTopo*, et ceux-ci utiliseront uniquement le numéro de cette sortie topo. Nous verrons plus tard comment gérer des cas plus complexes, mais pour l'instant, dans ce tutoriel, vous devriez pouvoir utiliser les noms de stations attribués automatiquement dans l'éditeur de cartes, ce qui rend le dessin beaucoup plus facile.

Ces fichiers représentant chaque séance peuvent éventuellement être placés dans leurs propres dossiers pour aider à mieux organiser les fichiers sur le disque, mais comme nous ne dessinons qu'une seule grotte dans ce tutoriel, ce n'est pas nécessaire. La grotte entière est alors représentée par un fichier maître .th qui contient les cartes qui rassemblent toutes les séances topographiques, et la définition de la ligne de cheminement qui joint toutes les lignes de cheminement pour l'ensemble des séances individuelles. Nous allons commencer cette leçon en réorganisant ce que nous avons dessiné jusqu'à présent dans ce tutoriel afin qu'il suive la structure de données décrite ci-dessus.

Jusqu'à présent, nous avons créé des fichiers de trajets nommés 'swildons' et 'OldGrotto'. Les fichiers de la séance Old Grotto sont déjà nommés 'OldGrotto.th' et 'OldGrotto.th2', ce qui correspond à ce que nous voulons pour cette séance. Mais les fichiers d'origine que nous avons créés dans les premières leçons de ce tutoriel portent le nom de la cavité. Nous devons donc déplacer les données de la topographie situés dans le fichier 'swildons.th' vers un nouveau fichier portant le nom de cette séance topo. Nous avons toujours besoin d'un fichier 'swildons.th' pour rassembler l'ensemble de la cavité. Nous allons donc COPIER le fichier 'swildons.th' et renommer cette copie en 'ShortDryWay.th', en laissant le fichier original 'swildons.th' où il était. Le fichier 'swildons.th2' contenant notre croquis ne concerne que la séance Short Dry Way, nous pouvons donc le renommer en 'ShortDryWay.th2'. Now ouvrez le fichier 'ShortDryWay.th' dans l'éditeur, et apportez les modifications suivantes :

- Changer le nom de la topographie en ShortDryWay.
- Modifier le nom du fichier .th2 dans la déclaration d'entrée pour qu'il corresponde au fichier .th2 renommé.
- Renommer la carte de 'swildonsMP' en 'ShortDryWayMP'.
- Supprimez la ligne : cs OSGB:ST.
- Supprimer l'entrée et les lignes de fixation.

Nos deux fichiers de données (OldGrotto.th et ShortDryWay.th) devraient maintenant contenir le même modèle d'éléments représentant un séance topo (ou une partie de la cavité). Nous pouvons traiter chaque séance séparément pour vérifier que tout fonctionne, en modifiant l'instruction 'source' dans le fichier *thconfig*. Nous allons conserver plusieurs déclarations de source dans ce fichier au fur et à mesure que nous travaillons sur le projet, mais une seule peut être active à la fois. Nous mettrons donc un caractère '#' au début des autres déclarations sources pour les commenter (ce qui signifie que *Therion* les ignorera). Vérifiez d'abord que le projet se compile avec la source toujours définie comme 'OldGrotto.th' pour confirmer que tout va bien. Vous devriez obtenir un PDF montrant la zone du Old Grotto.

La section suivante est importante à connaître si vous déplacez des fichiers dans différents dossiers et que vos croquis n'affichent plus le croquis de fond dans l'éditeur de carte. Mais si vous suivez ce tutoriel à la lettre, vous ne rencontrerez pas ce problème et vous pouvez sauter les 5 paragraphes suivants (passez

juste au-dessus du prochain graphique ci-dessous). Mais si vous voulez savoir comment réparer un arrière-plan cassé, lisez la suite (mais n'apportez pas de modifications à moins que vous n'ayez besoin de résoudre un problème !)

Ouvrez les fichiers 'ShortDryWay.th' et 'ShortDryWay.th2' dans l'éditeur. Si vous avez décidé d'organiser vos fichiers dans des dossiers séparés, lorsque vous regardez le croquis dans l'éditeur de carte, vous verrez que le dessin de fond n'apparaît plus. Cela est dû au fait que vous avez déplacé le fichier vers un nouveau dossier et que le chemin d'accès au dessin ne pointe plus vers le dossier contenant le dessin de l'arrière-plan. Vous pouvez éviter ce problème à l'avenir en décidant de la structure des dossiers de vos projets avant de créer les esquisses. Mais il est utile de savoir comment résoudre ce problème lorsque nous réorganisons nos fichiers de projet. Fermez le croquis dans l'éditeur de cartes et ouvrez le fichier .th2 dans un éditeur de texte normal, par exemple *Notepad* sous *Windows* ou *Notepad++* (voir les liens à la fin de ce tutoriel).

Vous verrez que même le fichier .th2 est en fait un fichier texte simple qui décrit tous les scraps, points et lignes que nous avons dessinés. Il est utile de le savoir car vous pouvez parfois retrouver et modifier le croquis dans un éditeur de fichiers texte pour résoudre des problèmes (comme la recherche de lignes ne contenant qu'un seul point). Les premières lignes du fichier codent l'emplacement du dessin de fond sur le disque, par rapport au fichier .th2. Notre fichier commence comme suit :

```
encoding    utf-8
##XTherion## xth_me_area_adjust -132.0 -132.0 1348.0 1192.0
##XTherion## xth_me_area_zoom_to 100
##XTherion## xth_me_image_insert {1028.81937007874 1 1.0}
{267.045118110236333.0} PocketTopo/ShortDryWayth_p.xvi 0 {}
```

La ligne '`image_insert`' est celle que nous devons modifier. Si nous avons déplacé notre fichier .th2 dans un sous-dossier, le dessin de fond (un fichier .xvi) n'est plus trouvé à partir de cet emplacement en utilisant le chemin '`PocketTopo/ShortDryWayth_p.xvi`'. Nous devons maintenant lui dire que le dossier *PocketTopo* se trouve un niveau plus haut. Ceci est indiqué en utilisant un code spécial pour 'le dossier un niveau plus haut' qui est une paire de points. Vous devez donc éditer cette ligne dans le fichier .th2 pour remplacer '`PocketTopo/ShortDryWayth_p.xvi`' par '`../PocketTopo/ShortDryWayth_p.xvi`'.

Sauvegardez le fichier, puis ouvrez-le à nouveau dans l'éditeur de cartes *Therion*. Vous devriez voir que le croquis de fond est maintenant affiché. Si tout cela vous semble trop compliqué, vous pourriez plutôt déplacer tous les fichiers liés à un croquis dans le même dossier que le croquis. Ensuite, si vous réorganisez vos dossiers à l'avenir, les chemins d'accès devraient fonctionner tant que vous conservez tous les fichiers de la séance topo dans le même dossier. Si vous décidez de procéder ainsi, vous devrez modifier les fichiers .th2 que vous avez déjà créés afin que le chemin d'accès au croquis d'arrière-plan soit simplement le nom du fichier .xvi. Ainsi, dans notre cas, nous remplacerons le fichier '`PocketTopo/ShortDryWayth_p.xvi`' par '`ShortDryWayth_p.xvi`', puis nous conserverons le fichier `ShortDryWayth_p.xvi` dans le même dossier que le fichier `ShortDryWay.th2`. N'oubliez pas de corriger le chemin du fichier .xvi dans le fichier `OldGrotto.th2` également, et vérifiez que le croquis de fond apparaît lorsque vous l'ouvrez dans l'éditeur de cartes *Therion*.



C'est ici qu'il faut reprendre les choses si vous avez sauté cette dernière section et que vos dessins affichent leurs croquis de fond dans l'éditeur de cartes. Ouvrez à nouveau le croquis de Short Dry Way dans l'éditeur de cartes. Nous avons initialement nommé le calque créé dans les leçons précédentes "swilSP1". Sélectionnez le scrap dans l'éditeur de cartes (utilisez le bouton "Sélectionner le scrap suivant").



Changez ensuite le nom du calque (scrap) dans le panneau d'information "scrap" en "ShortDryWaySP1". Nous devons également modifier le nom du scrap dans la définition de la carte dans le fichier 'ShortDryWay.th' pour qu'il corresponde. Maintenant, commentez la ligne source pour la Vieille Grotte dans le fichier *thconfig*, et ajoutez une nouvelle ligne source pour notre séance sur Short Dry Way:

```
#source OldGrotto.th source  
ShortDryWay.th
```

Maintenant, lorsque vous compilez le projet, vous devriez voir que le PDF contient la séance Short Dry Way au lieu de celle d'Old Grotto. Nous avons maintenant un projet dans lequel nous pouvons travailler sur le croquis de l'une ou l'autre de nos deux séances topo, et ajouter de nouvelles séances en suivant le même modèle. Il ne reste plus qu'à créer une source qui combine les diverses séances en un seul plan-carte.

Ouvrez le fichier original 'swildons.th' dans l'éditeur. Nous ne voulons conserver que les informations qui n'ont pas fait partie d'une séance topo spécifique (la déclaration du système de coordonnées et le repère d'entrée).

Voici les modifications que nous allons apporter à ce fichier :

- Supprimer la déclaration d'entrée pour l'ancien fichier .th2.
  - À la place, nous voulons entrer ShortDryWay.th et OldGrotto.th ici.
- Retirer la date, l'équipe et toutes les lignes de données du relevé du bloc de ligne de cheminement (seules la déclaration du système de coordonnées, l'entrée et le repère doivent rester dans la ligne centrale).
- Renommer l'entrée et la station fixe en " ent ".
- Ajouter des déclarations pour relier notre station d'entrée à une station d'une de nos séances topo, et pour relier nos deux séances entre elles:
  - equate ent 3.0@ShortDryWay
  - equate 3.13@ShortDryWay 4.0@OldGrotto
- Supprimer la vieille référence de scrap de la carte, et la remplacer par les cartes de nos séances topo.
  - ShortDryWayMP@ShortDryWay
  - OldGrottoMP@OldGrotto

Notez que maintenant, lorsque nous faisons référence à un élément dans l'un de nos fichiers de séances topo nous devons inclure le nom de la topographie dans la référence. Le format de ces références est le suivant :

```
<object name> @ <survey name>
```

Ainsi, pour faire référence à la station 3.0 dans la topo Short Dry Way, nous utilisons la référence : 3.0@ShortDryWay

Il en va de même pour les cartes. Ainsi, dans notre carte principale, nous faisons référence à la carte de la topo Old Grotto en utilisant la référence : OldGrottoMP@OldGrotto

La déclaration de carte dans notre fichier swildons.th contient maintenant les cartes qui sont définies dans chacune de nos séances topo. Les cartes peuvent contenir soit des "scraps" soit des "maps", mais vous ne pouvez pas mélanger les deux dans la même carte. C'est pourquoi nous définissons une carte dans chacun de nos fichiers .th pour les séances topo individuelles, même s'il n'y a qu'un seul scrap par séance. De cette façon, nous pouvons combiner toutes les cartes en une seule carte principale.

Maintenant, nous devrions être en mesure de compiler l'ensemble du projet de la grotte. Changez la source dans le fichier *thconfig* en 'swildons.th' et compilez le projet. Le PDF devrait maintenant montrer tous les calques que nous avons dessinés sur un seul plan. Comme nous avons inclus un système de coordonnées et fixé une station dans le relevé à un emplacement réel (géoréférencé), nous pouvons ajouter les instructions d'exportation pour les fichiers kml.

Enfin, comme nous avons maintenant inclus des calques (scraps) qui sont joints bout à bout à partir de différents relevés en plan, nous devons "joindre" les calques pour assurer des joints réguliers entre les extrémités des parois. Nous avons délibérément déplacé l'un des points d'extrémité de la ligne dans le scrap ShortDryWaySP1 pour montrer une extrémité de paroi mal alignée dans les fichiers d'exemple de cette leçon. Vous verrez donc un écart entre les extrémités des parois à l'endroit où ce fragment de notre relevé ShortDryWay rejoint notre relevé OldGrotto. Il s'agit d'une situation courante où les dimensions de la galerie ont été estimées plutôt que mesurées. Nous pouvons corriger cet écart en demandant à *Therion* de joindre ces fragments. Nous devons déclarer cette jointure dans notre fichier swildons.th parce qu'elle implique des fragments de plus d'une séance de relevé topo. N'oubliez pas de référencer les scraps en incluant les noms des séances topo. Placez la déclaration de jointure juste après les lignes 'input'.

Voici ce que contient maintenant le fichier swildons.th complet :

```
survey swildons

input ShortDryWay.th input
OldGrotto.th

join ShortDryWaySP1@ShortDryWay ShortDryWaySP2@OldGrotto

map swildonsMP
  ShortDryWayMP@ShortDryWay
  OldGrottoMP@OldGrotto
endmap

centreline
  cs OSGB:ST

station ent "main ent." entrance
fix ent 53122 51308 237

equate ent 3.0@ShortDryWay
```

```
equate 3.13@ShortDryWay 4.0@OldGrotto  
endcentreline  
endsurvey
```

Nous disposons maintenant d'une structure de données qui peut être étendue à d'autres séances topo pour constituer une topographie plus importante. Dans la prochaine leçon, nous verrons comment améliorer l'apparence de notre topographie en utilisant des "mises en page" pour personnaliser la présentation de notre dessin.

## Leçon 9 : Utiliser les mises en page (layouts) pour changer de style

Jusqu'à ce point, nous avons dit à *Therion* ce qu'il devait dessiner, mais pas comment le faire. *Therion* a choisi comment restituer tous les différents types de lignes et de points sur notre relevé et quelles tailles et polices utiliser pour les étiquettes (nous avons seulement spécifié un facteur d'échelle en utilisant xs, s, m, l, xl). Nous pouvons contrôler tous ces éléments, ainsi que les couleurs utilisées. Nous pouvons également contrôler ce qui apparaît sur la page et où.

Par défaut, *Therion* affiche le nom du relevé, la longueur, la profondeur, les noms des topographes et une barre d'échelle. Dans cette leçon, nous allons apprendre à modifier tous ces éléments, y compris leur emplacement sur la page.

Lorsque vous commencez à apprendre *Therion*, vous êtes confronté au dilemme suivant : vous devez dessiner une partie suffisante du relevé pour voir comment les tailles des symboles et du texte apparaissent lorsque vous les modifiez. Mais il ne faut pas trop dessiner avant de décider des tailles, car les tailles et les symboles que vous choisirez changeront comment et où vous voulez les placer. Maintenant que nous avons dessiné une partie suffisante de notre topo pour avoir une bonne idée, il est temps de décider de l'échelle à laquelle nous voulons produire la topographie, et donc des tailles à utiliser pour les étiquettes et les symboles. Si vous dessinez une grotte de 30 mètres de long pour qu'elle tienne sur une feuille de papier A3, vous pouvez utiliser des symboles et des étiquettes assez petits et entrer beaucoup de détails. Mais si vous dessinez une grotte de 15 km de long pour la faire tenir sur une grande feuille de papier, vous aurez besoin de symboles et d'étiquettes beaucoup plus grands pour qu'ils soient lisibles sur la feuille imprimée (ou sur un écran avec un zoom arrière suffisant pour voir une grande partie de la grotte).

Nous allons commencer par définir une mise en page où nous pourrions commencer à modifier certaines choses. Ceci est fait dans le fichier *thconfig*, donc passez à la fenêtre 'Compiler', et ajoutez ce qui suit en dessous de vos instructions 'export' :

```
layout local endlayout
```

Nous pouvons utiliser n'importe quel nom pour notre layout (en fait, vous pouvez avoir plusieurs définitions de layout et les combiner à des fins différentes). Mais la convention que j'utilise est d'appeler la mise en page " local " lorsqu'elle est définie dans le fichier que je compile pour le projet. Maintenant que nous avons défini une mise en page (même si elle n'a pas encore de contenu), nous pouvons demander à *Therion* de l'appliquer dans notre déclaration d'exportation PDF. Ajoutez le texte 'layout local' dans la commande

d'exportation du plan PDF comme suit :

```
export map -projection plan -layout local -o swildons.pdf
```

Vérifiez que le projet se compile toujours bien. Vous ne devriez pas voir de changement, mais cela confirme que tout a été fait correctement. Maintenant que nous avons défini une mise en page, nous pouvons commencer à la personnaliser. Nous allons commencer par les couleurs d'arrière-plan. Il y a deux zones principales pour lesquelles nous pouvons définir des couleurs, l'arrière-plan de la page (toute la page à l'extérieur de notre grotte) et la galerie de la grotte (à l'intérieur des parois de notre cavité). Les couleurs sont définies à l'aide du système RGB, où trois nombres sont déclarés pour chacune des composantes rouge, verte et bleue. Dans *Therion*, ces nombres sont compris dans la plage 0 à 100. Le rouge est déclaré comme [100 0 0], le vert comme [0 100 0] et le bleu comme [0 0 100]. Lors de la déclaration d'une couleur, il est utile d'inclure un commentaire donnant un nom descriptif à la couleur, afin que l'on sache clairement à quoi elle ressemblera dans le fichier de configuration.

*Therion* désigne l'arrière-plan de la page par `map-bg` et l'intérieur de la grotte par `map-fg`. Ajoutez la ligne suivante dans votre définition de layout et compilez le projet pour voir la galerie de la grotte colorée avec un remplissage jaune pâle :

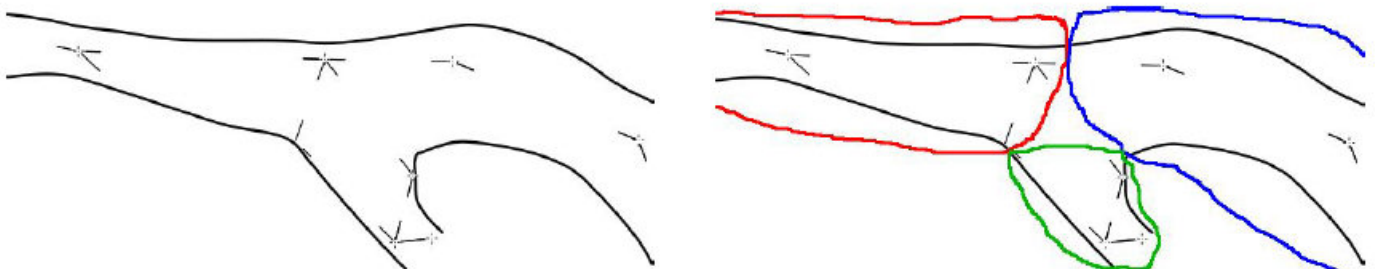
```
color map-fg [97 86 38] # pale yellow
```

Nous pouvons colorer le fond de la page en gris en ajoutant la ligne suivante dans la mise en page :

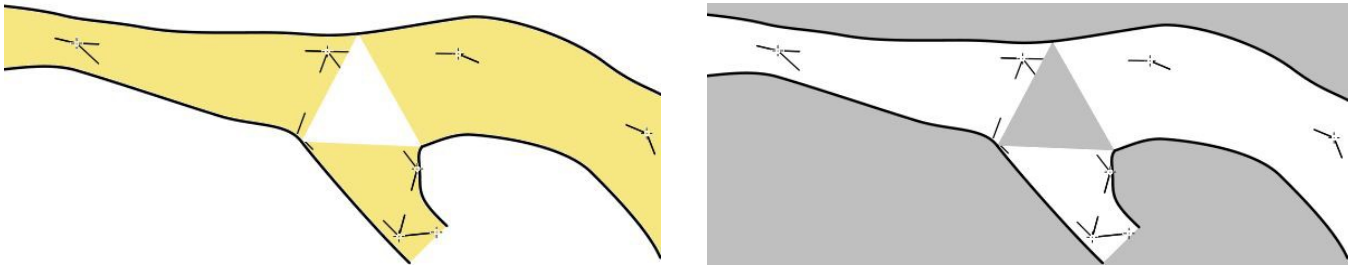
```
color map-bg [75 75 75] # light-grey
```

Maintenant que les calques (scraps) sont rendus avec un fond coloré, nous pouvons voir s'il y a des lacunes dans nos jonctions de calques. C'est une bonne idée d'utiliser une couleur de carte lors du dessin, car vous repérerez des problèmes qui pourraient autrement être cachés jusqu'à ce que vous visualisiez le modèle ou la sortie KML. Un exemple typique est le raccordement de trois calques (scraps) à une jonction. Comme les bords des calques (scraps) sont une ligne droite à travers les extrémités des parois, il est possible que cela laisse un trou dans la grotte si vous choisissez le mauvais endroit pour terminer vos calques.

Voici un exemple : la galerie de la grotte a l'air bien présentée dans un simple PDF en noir et blanc.

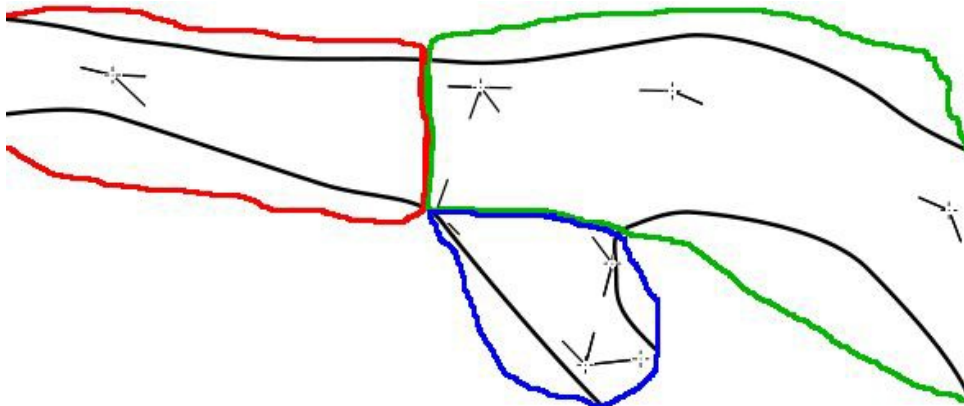


L'image de droite montre l'étendue des parois des 3 calques (scraps) reliés à cette jonction.



Si nous définissons une couleur `map-fg` jaune pâle, un trou dans le modèle apparaît.  
 Nous pouvons également le voir clairement si nous définissons un fond de page gris.

Lorsque vous joignez des scraps près d'une jonction, évitez d'avoir trois scraps tous joints au même endroit, en déplaçant les joints dans les galeries sur le côté de la jonction. Ainsi, toute la zone située à l'intérieur des parois de la grotte fait partie de l'un des morceaux :

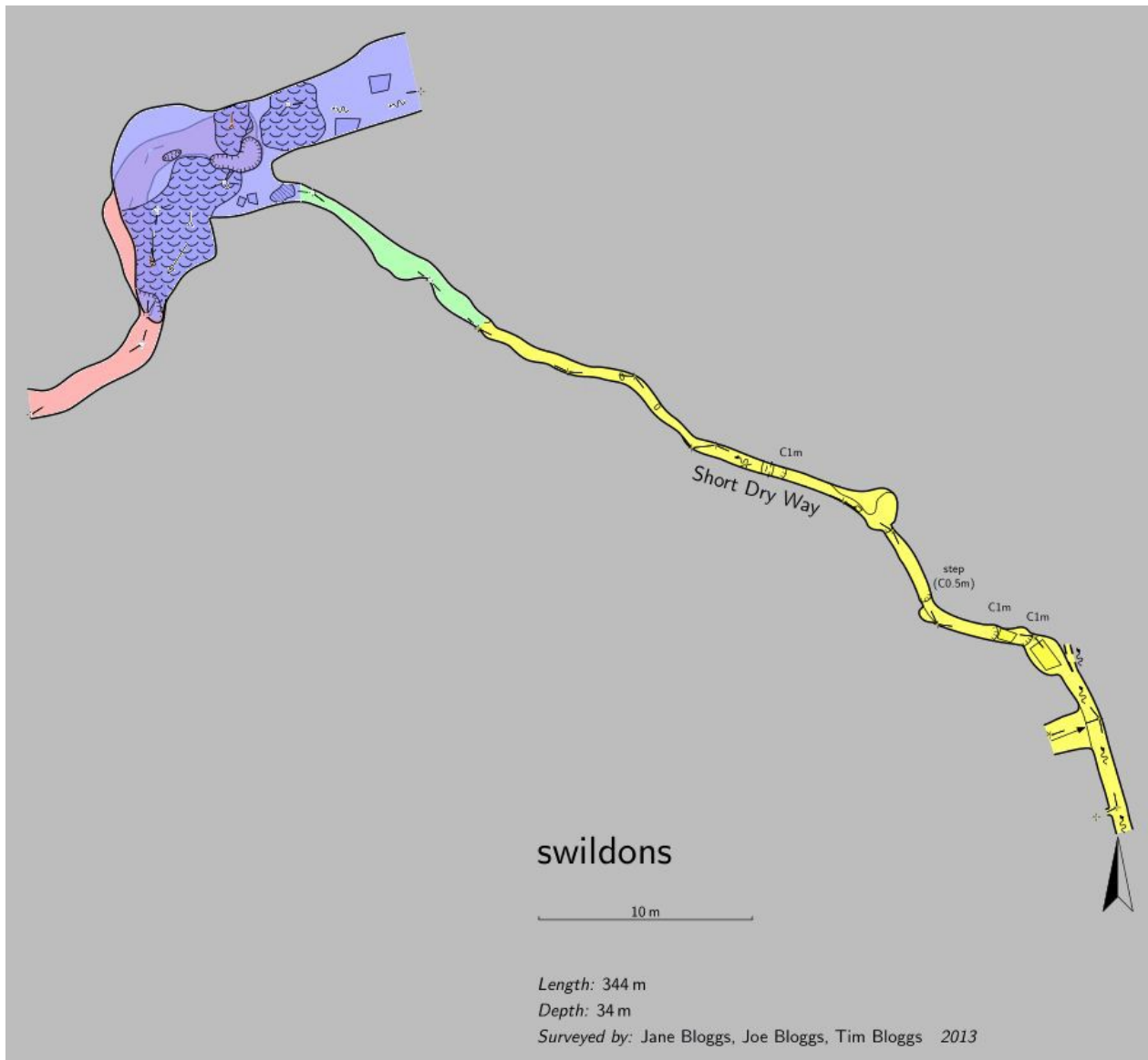


Une meilleure façon de joindre 3 scraps (délimités par des couleurs) à une jonction

Vous pouvez également demander à *Therion* d'utiliser des couleurs différentes pour chaque scrap (ou carte) afin de mieux les distinguer dans votre topographie. Modifiez la commande `map-fg` dans la mise en page comme suit pour donner une couleur différente à chaque scrap :

```
color map-fg scrap
```

Ceci est très utile lorsque vous travaillez sur le dessin pour voir comment les différents morceaux sont superposés les uns sur les autres, comme vous pouvez le voir dans notre exemple utilisant cette option (avec une couleur `map-bg` gris pâle).

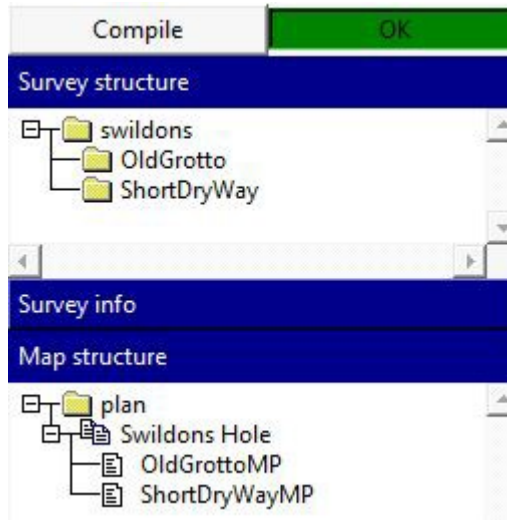


Maintenant que nous avons mis un peu de couleur (ou de gris si vous préférez) dans le relevé topo, il est temps de regarder la boîte titre et les informations affichées. Nous n'avons pas encore dit à *Therion* ce que nous voulions qu'il génère comme plan, donc le programme a choisi la première définition de carte qu'il a trouvée dans le fichier source .th que nous avons spécifié dans le fichier *thconfig*. Nous n'avons pas encore donné de titre aux cartes, nous allons donc le faire maintenant afin que *Therion* sache comment nous voulons appeler notre plan. Ouvrez le fichier *swildons.th*, et ajoutez un titre à la définition de carte *swildonsMP* comme suit :

```
map swildonsMP -title "Swildons Hole"
```

Si vous compilez à nouveau le projet, vous remarquerez qu'il n'affiche toujours pas le titre. En fait, il n'affiche même pas le nom de la carte "swildonsMP". La boîte de titre actuelle affiche le nom trouvé dans le fichier topo. Nous devons choisir spécifiquement la carte que nous voulons tracer pour que *Therion* puisse afficher le titre de la carte. Une fois que nous avons compilé le projet au moins une fois, la fenêtre du compilateur (appuyez sur F3) aura rempli certains champs dans le panneau de droite avec la structure de

notre projet. Dans notre cas, nous devrions maintenant voir les cases structure de la topographie et structure de la carte remplies :



Remarquez qu'il a déjà pris le nouveau titre 'Swildons Hole' que nous avons donné à notre carte swildonsMP, alors que les autres cartes affichent toujours le nom de la carte. Cliquez dans le panneau d'édition du fichier *thconfig* pour placer le curseur en haut du fichier, puis double-cliquez sur le nom de la carte 'Swildons Hole'. L'éditeur devrait insérer une instruction de sélection pour cette carte dans le fichier de configuration.

```
select swildonsMP@swildons
```

Il a utilisé le chemin complet vers cette carte dans la topographie. Maintenant que *Therion* sait quelle carte nous voulons utiliser, il affichera le titre de cette carte dans le plan lorsque vous compilerez le projet. Vous devriez donc maintenant voir le titre 'Swildons Hole' affiché sur le plan topo.

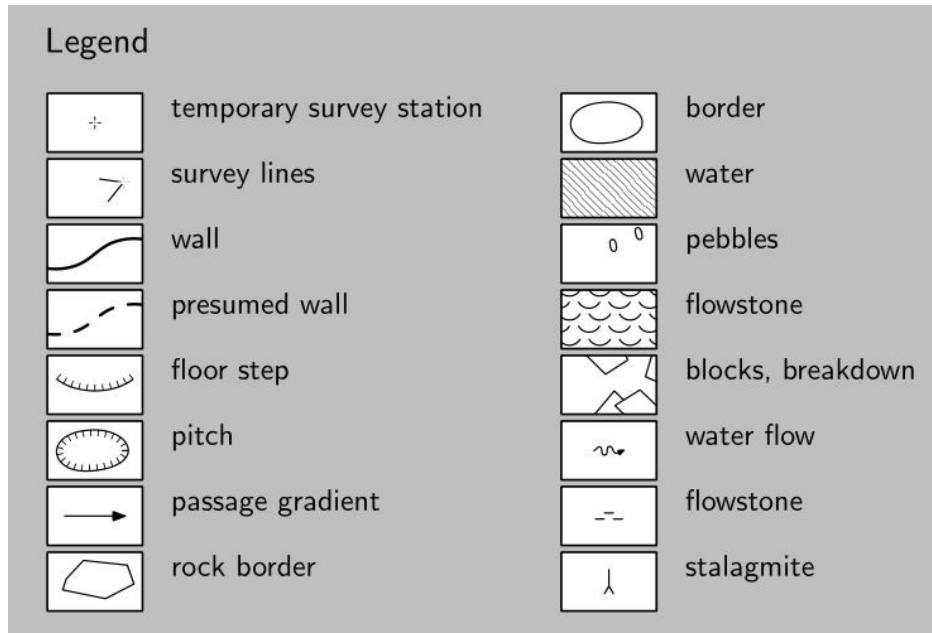
Notez que vous ne pouvez utiliser qu'une seule instruction `select` dans un fichier *thconfig*, et l'endroit où vous la placez dans le fichier importe peu. Je la place toujours en haut du fichier pour qu'elle soit facile à repérer. Si vous placez des instructions de sélection pour plus d'une carte, seule l'une d'entre elles sera utilisée. Ainsi, pour générer différents PDF à l'aide de différentes cartes, vous devez créer plusieurs fichiers *thconfig* pour votre projet. Vous pourriez typiquement faire cela pour générer un PDF à grande échelle de la série d'entrées de la grotte, et une carte à plus petite échelle de la grotte entière. Nous verrons comment modifier l'échelle plus tard dans cette leçon.

Ensuite, nous allons ajouter quelques informations supplémentaires à la boîte de titre. Pour l'instant, elle affiche la liste des membres de l'équipe et la date des séances topo, une barre d'échelle, une flèche nord et la longueur et la profondeur de la grotte. Si vous changez la couleur de `map-fg` en "altitude", vous verrez également une clé pour les couleurs. Celle-ci est générée automatiquement à partir de la gamme d'altitude des stations de votre relevé dans les calques (scrap) que vous avez inclus dans la carte. Il n'y a actuellement aucun moyen de modifier les couleurs utilisées, mais vous pouvez déterminer l'altitude minimale et maximale en incluant dans votre projet quelques fausses stations de mesure fixées à l'altitude minimale et maximale que vous souhaitez colorier. Notez que chaque scrap est d'une seule couleur, donc l'altitude pour le scrap est basée sur toutes les stations du relevé topo de ce scrap. Si vous souhaitez générer des relevés en utilisant la couleur en fonction de l'altitude, il est fortement recommandé de commencer un nouveau scrap chaque fois qu'il y a un changement significatif de hauteur (par exemple, sur un terrain), ou lorsque le changement de hauteur du sol sur le scrap est de plus de 5-10 mètres.

Nous avons déjà utilisé un grand nombre de symboles dans notre dessin, il serait donc utile d'inclure une légende. Pour ce faire, il suffit d'ajouter la ligne suivante dans la définition de la mise en page :

```
legend on
```

Maintenant, lorsque vous compilez le projet, le PDF devrait afficher une légende illustrant chaque symbole utilisé dans le dessin.



À ce stade, notre boîte de titre et notre légende occupent plus de la moitié de la page, ce qui n'est probablement pas ce que vous souhaitez pour une petite topographie comme celle-ci. Vous pouvez modifier cela en changeant l'échelle. L'échelle par défaut est adaptée aux cavités plus importantes où il y a beaucoup plus de galeries à faire tenir sur la page. Mais pour la petite section de grotte de notre carte tutorial, il est plus logique de dessiner la galerie plus grande sur la page. Ajoutez la ligne suivante dans la définition de la mise en page :

```
scale 1 100
```

La galerie de la grotte sera ainsi plus grande sur la page par rapport à la taille de la légende. Notez que les symboles et le texte n'ont pas été agrandis, seule l'échelle utilisée pour tracer la position de la galerie a changé. En conséquence, le texte, les symboles et la légende apparaissent beaucoup plus petits à l'écran lors de la visualisation du PDF complet (en fait, ils ont gardé la même taille, mais la taille globale de la page a augmenté pour s'adapter au dessin à plus grande échelle des passages souterrains). Maintenant que les galeries sont deux fois plus grandes qu'auparavant (l'échelle par défaut était de 1:200), le symbole contient beaucoup plus de cailloux ou de blocs à l'intérieur des endroits sélectionnés. Vous pouvez contrôler l'échelle utilisée pour dessiner les symboles et les étiquettes de la topographie séparément si nécessaire. Si vous souhaitez les ramener à l'échelle 1:200 tout en conservant les galeries à l'échelle 1:100, vous pouvez déclarer l'échelle de base en même temps que l'échelle :



```
scale 1 100
base-scale 1 200
```

Notez que les symboles et le texte des étiquettes ont maintenant doublé de taille, mais que la légende est toujours de la même taille. Vous pouvez également contrôler la taille des étiquettes de texte individuellement avec leur option d'échelle dans l'éditeur de carte (rappelez-vous que nous avons défini l'option "scale l" lors de l'ajout des points d'étiquette). Il existe 5 codes d'échelle intégrés pour les étiquettes : `xs`, `s`, `m`, `l`, `xl` qui peuvent être utilisés pour contrôler la taille relative du texte.

Le type et la taille de la police pour chacun de ces codes d'échelle peuvent également être définis, tout comme l'apparence des symboles. Mais cela sera abordé dans la prochaine leçon. Le point principal à retenir ici est que l'apparence de la topographie est fortement contrôlée par les paramètres de mise en page que vous utilisez pour la restituer. Par conséquent, avant de dessiner un grand nombre d'éléments, vous devez décider de l'échelle et des styles de symboles que vous souhaitez pour votre résultat. Sinon, vous vous retrouverez à devoir redessiner beaucoup de choses lorsque vous constaterez que les symboles ne tiennent plus à l'endroit où vous les avez placés, ou que le texte chevauche d'autres détails de la topographie et doit être repositionné.

Il existe quelques autres éléments que nous pouvons ajouter à notre mise en page. Nous pouvons ajouter un commentaire en texte libre en utilisant la commande de commentaire de carte dans la mise en page. Par exemple, nous pouvons ajouter un message de copyright :

```
map-comment "Copyright 2016 Footleg Caving Expedition"
```

Une grille est également utile. Pour cela, nous devons spécifier la référence nord de la grille (nous devons toujours le faire, que nous choissions de montrer une grille ou non, car cela s'applique également à la flèche du nord). Nous avons déjà défini un système de coordonnées dans notre topographie, nous allons donc spécifier que la référence nord doit être "grid" (l'alternative est "true").

Nous devons préciser si le quadrillage de la grille est dessiné au-dessus des galeries de la grotte, ou en dessous sur la carte, en utilisant 'top' ou 'bottom' dans la déclaration de la grille. Nous devons également spécifier l'espacement de la grille.

Cet espacement est tridimensionnel car nous pouvons également dessiner une carte en coupe, ce qui signifie que la dimension verticale est appliquée lors du dessin d'une grille. Sur notre plan, seuls les espacements de grille `Easting` et `Northing` affecteront la grille. Notre étude est de petite taille, nous allons donc spécifier une grille de 10m x 10m pour le plan (et 10m pour la dimension verticale également, par souci de cohérence).

```
north grid grid
bottom
grid-size 10 10 10 m
```

Vous devriez maintenant voir une grille sur le relevé. Nous pouvons ajouter des coordonnées de grille imprimées le long des frontières seulement, ou à chaque intersection de grille, en utilisant 'all' ou 'border' :

```
grid-coords all
```

Enfin, vous ne souhaitez peut-être pas que le Nord soit en haut de la page dans la carte. Vous pouvez changer cela en utilisant la commande `rotate`. Cette commande fait pivoter la page du nombre de degrés spécifié. Ainsi, la commande suivante fera en sorte que le Nord soit placé à un angle de 45 degrés sur la page :

```
rotate 45
```

Enfin, nous pourrions souhaiter déplacer la boîte de titre sur la page afin qu'elle s'adapte mieux à la topographie. Actuellement, il étend la taille de la page en dessous de la topographie, laissant beaucoup d'espace vide sur la page.

Nous pouvons contrôler la position en utilisant la commande `map-header` dans la mise en page. Cette commande spécifie où placer un coin de la boîte d'en-tête. La position est donnée en termes de pourcentage de la largeur du relevé et de la hauteur du relevé (à partir du coin inférieur gauche).

La référence du coin ou du bord de la boîte de titre est spécifiée comme une direction de la boussole (`n, ne, e, se, s, sw, w, nw`). Ainsi, si vous souhaitez placer le coin nord-est de la boîte de titre dans le coin supérieur droit du levé, vous utiliserez cette commande :

```
map-header 100 100 ne
```

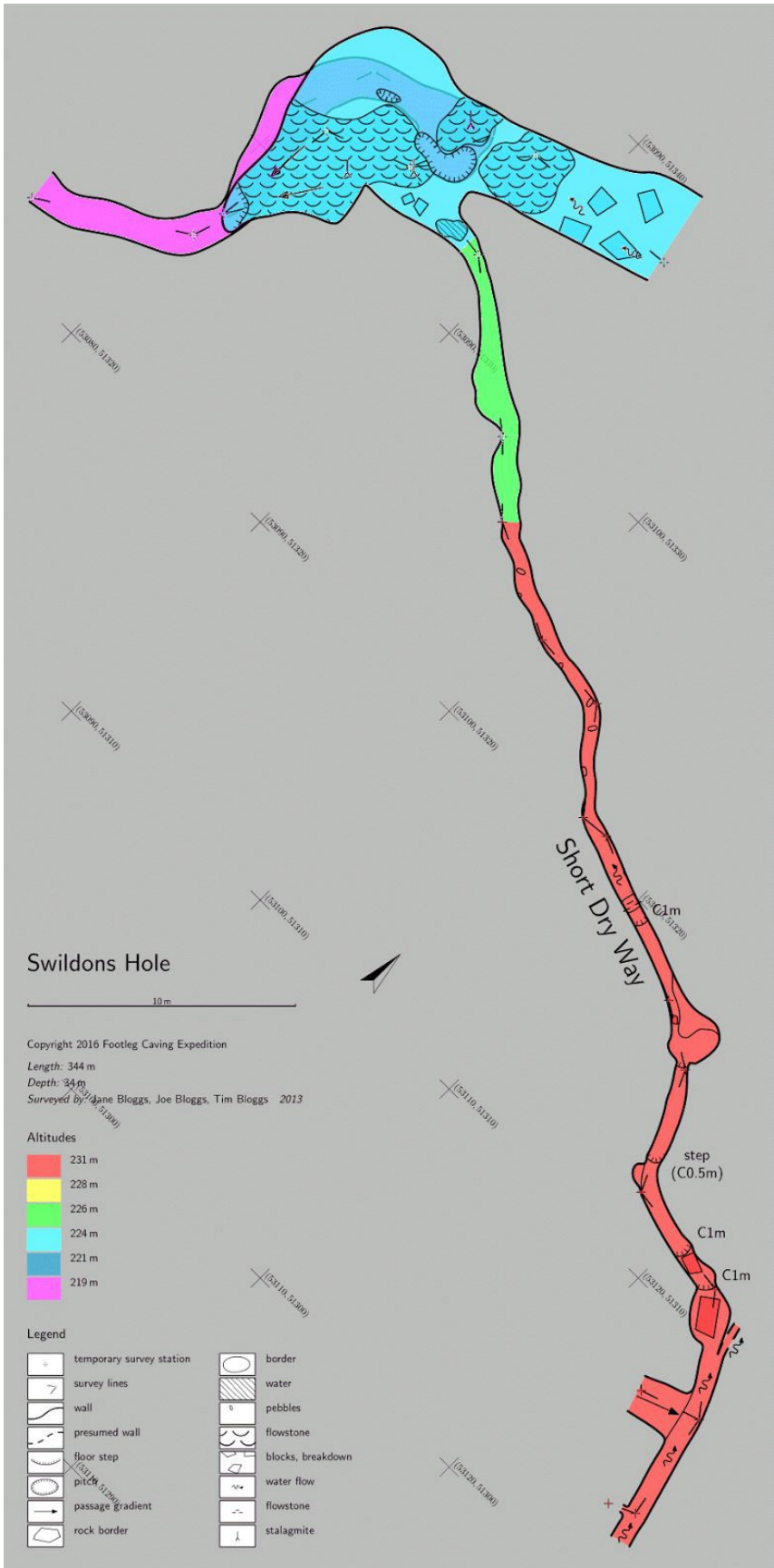
Notez comment cela fait chevaucher la grotte sur notre page. Pour placer la boîte d'en-tête à droite de la topographie, nous voulons positionner le coin NW de la boîte d'en-tête dans le coin supérieur droit de la topographie :

```
map-header 100 100 nw
```

Dans notre exemple de tracé, il est plus logique de le placer à l'intérieur de la zone du tracé de la topographie, mais dans l'espace vide du coin inférieur gauche. Nous allons donc placer le coin sud-ouest de la boîte d'en-tête dans le coin inférieur gauche de la topographie :

```
map-header 0 0 sw
```

Il y a beaucoup d'autres choses qui peuvent être personnalisées dans les layouts. Pour plus d'informations, consultez la section " layouts " du *Therion Book* PDF (p.49). Dans la prochaine leçon, nous verrons comment personnaliser les symboles eux-mêmes et comment utiliser les mises en page prédéfinies développées par d'autres. A la fin de cette leçon, voici à quoi ressemble notre plan topo final :



## Leçon 10 : Contrôler les styles de symboles et utiliser des mises en page prédéfinies

Cette leçon abordera la personnalisation des symboles utilisés, le masquage des marqueurs d'arpentage et la personnalisation des échelles de texte.

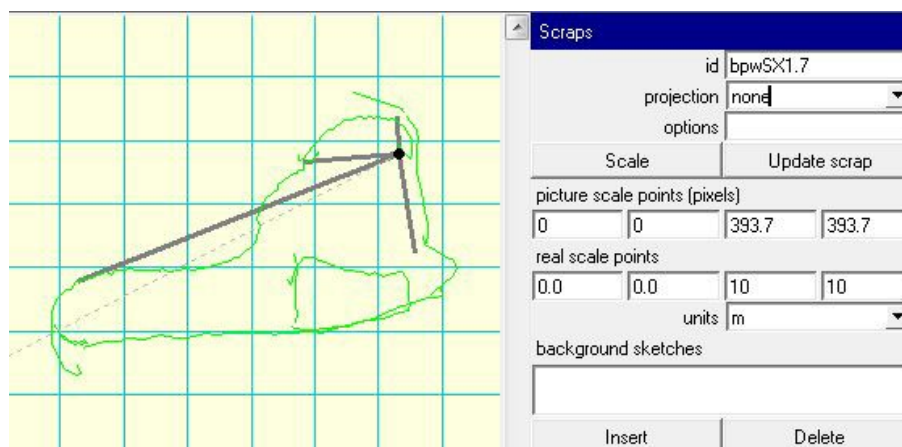
(Commencez par une grotte plus dessinée dans cette leçon, afin que les styles et la mise en page mettent en valeur davantage de caractéristiques).

## Leçon X : Ajouter des sections transverses

(Note : Cette leçon utilise toujours un exemple d'un ancien tutoriel, et doit être mise à jour pour couvrir ce sujet en utilisant un exemple de Swildons)

Dans cette leçon, nous allons découvrir un nouveau type de scrap, utilisé pour ajouter des sections transverses à notre topographie. Chaque section est dessinée dans un scrap séparé, et ces sections sont ensuite incorporées dans la topographie en utilisant un type de point spécial.

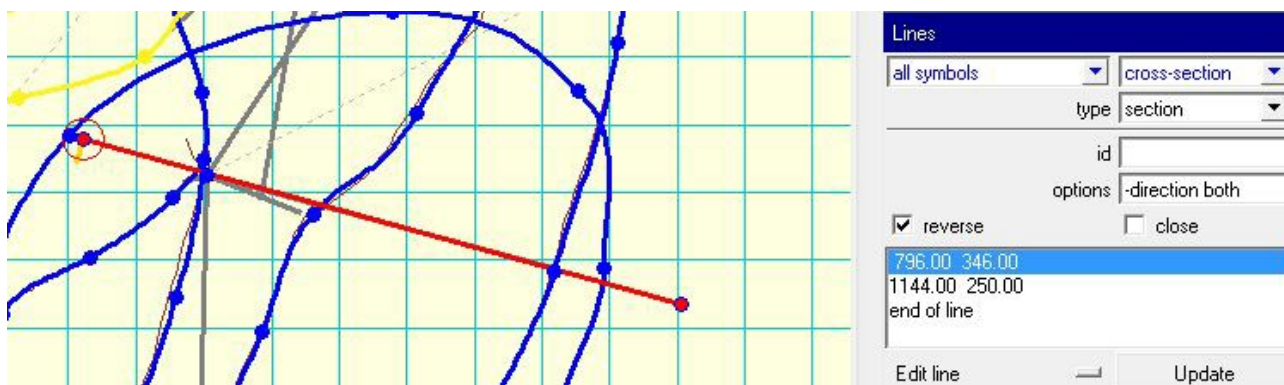
Tout d'abord, nous devons créer un nouveau scrap pour dessiner l'une des sections transverses. Nous allons dessiner la section à la station 1.7, donc nous identifierons le scrap en utilisant le nom de notre levé, suivi de 'SX' pour scrap+xsection. Comme nous aurons un scrap de section X pour chaque section dessinée, nous devons également ajouter le nom de la station pour que l'identifiant de chaque scrap soit différent de tous les autres. Définissez donc le champ id pour le nouveau scrap comme étant 'bpwSX1.7'. Ce scrap n'est ni un plan ni un dessin de coupe, nous devons donc définir le champ projection sur "none". Voici la section que nous allons dessiner, et le panneau d'information du scrap.



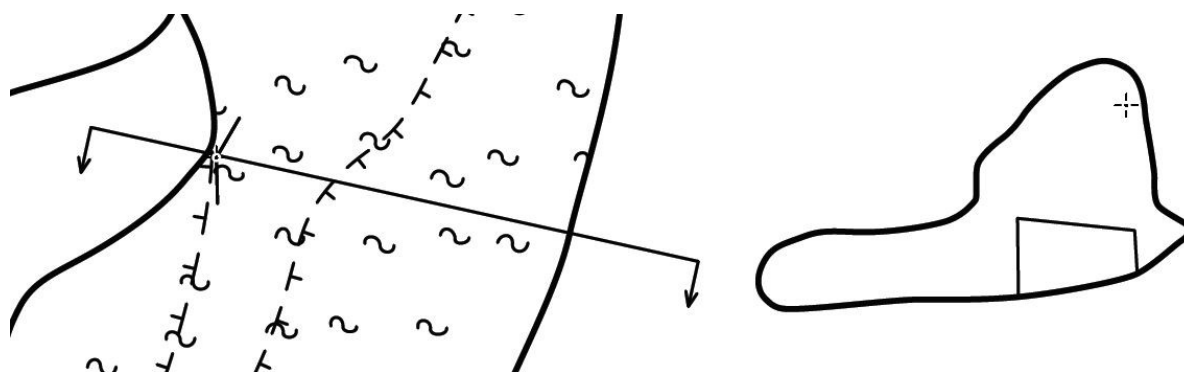
Notez qu'un scrap de xsection ne contient qu'une seule station topo, donc *Therion* ne peut pas calculer l'échelle à partir de la distance entre les stations. C'est là que les informations sur l'échelle dans le panneau d'information du scrap deviennent cruciales. Ici, les valeurs ont déjà été saisies correctement car nous avons défini l'échelle lorsque nous avons créé le premier scrap à partir du dessin *PocketTopo* dans la leçon 3. Tous les nouveaux scraps seront créés avec la même échelle que le dernier scrap dans le même fichier .th2, c'est pourquoi nous l'avons configuré dès le début. Maintenant, nous pouvons dessiner notre section. Utilisez les lignes de paroi pour le contour, et les lignes de bordure de rocher pour le rocher. Il n'est pas nécessaire d'ajouter la station d'arpentage à la chute, mais cela permet d'indiquer la position de la station sur la section dans le dessin d'arpentage final. Cela permet également d'associer clairement la section à la station dans les données.

Maintenant que nous avons dessiné la section, nous devons l'inclure dans la topographie. Pour ce faire, nous indiquons le plan de la section sur l'un de nos morceaux de plan. Sélectionnez donc le morceau contenant le plan de la partie de la grotte où se trouve notre section. Tracez maintenant une ligne droite à travers la galerie de la grotte pour indiquer le plan de la section. Pour ce faire, cliquez sans faire glisser la souris afin

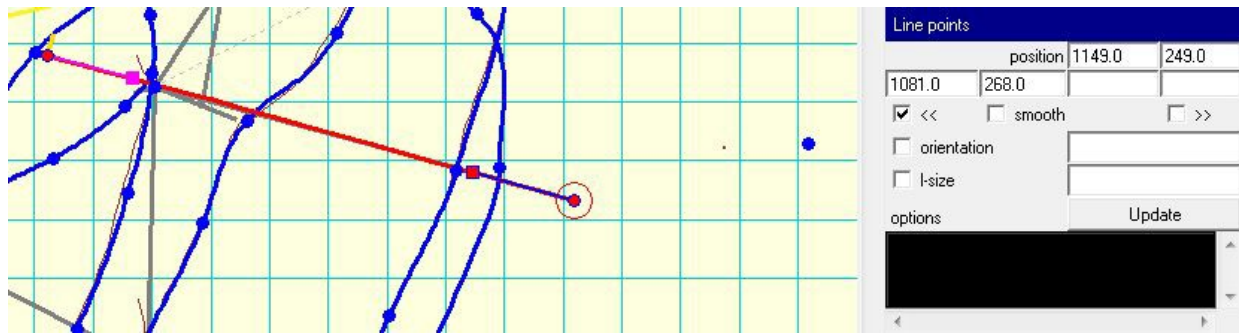
d'obtenir une ligne droite. Changez le type de ligne en "section". Assurez-vous que la coche jaune sur la ligne pointe dans la direction vers laquelle notre section a été dessinée. Dans notre cas, il s'agit du bas de notre esquisse. Nous avons des flèches à chaque extrémité de la ligne de section sur le dessin final, ou aux deux extrémités. Nous choisissons ici les deux, et nous devons donc ajouter le texte "direction both" dans le champ d'options de la ligne. Si nous ne voulions qu'une flèche à une extrémité, nous utiliserions "direction begin" ou "direction end" pour indiquer quelle extrémité. direction end' pour indiquer à quelle extrémité de la ligne la flèche doit être dessinée.



Maintenant, nous devons ajouter un point pour montrer où nous voulons que la section x réelle apparaisse sur le plan. Ajoutez un point près de l'extrémité de la ligne de section et définissez le type de point sur 'section'. Nous devons spécifier le scrap de section que nous voulons voir dessiné à cette position, donc définissez le champ d'options pour le point de section sur 'scrap bpwSX1.7'. Cela va dessiner notre section sur le plan à la position où nous avons placé le point. Voici à quoi ressemble maintenant le plan PDF.

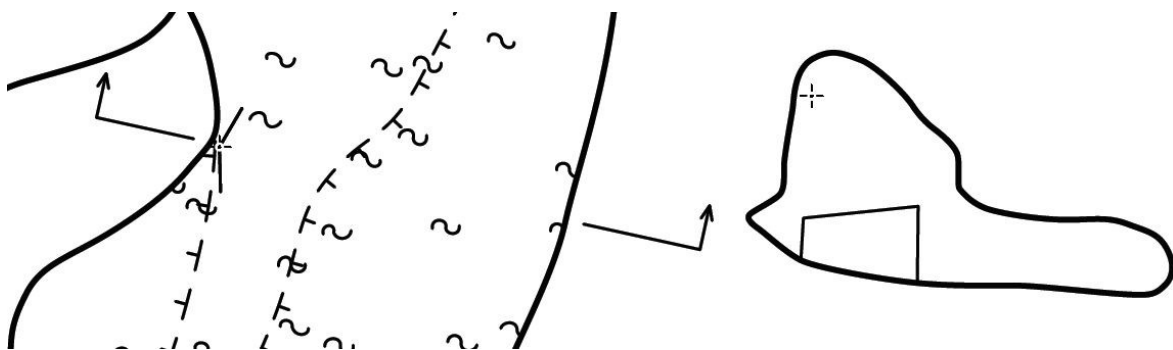


Normalement, nous devrions dessiner une section en regardant vers le haut de la page, de sorte que la section telle que dessinée sur la page ait la même orientation gauche-droite que la galerie sur le plan. Dans notre cas, c'est l'inverse, car dans la grotte, nous avons dessiné ces sections en arpentant le sud (c'est-à-dire en regardant vers le bas de la page). Nous pouvons facilement corriger cela dans *Therion*. Il suffit d'ajouter l'option 'flip horizontal' au scrap de la section. Sélectionnez donc le scrap où la section a été effectivement dessinée (bpwSX1.7) et ajoutez cette option au champ d'options dans le panneau d'information du scrap. Nous devons également inverser la direction de la coche sur notre ligne de section afin que les flèches aux extrémités pointent maintenant vers le haut de la page.



Il existe une autre astuce intéressante que nous pouvons utiliser avec la ligne de coupe. Si nous ajoutons des points de contrôle de courbe aux points situés à chaque extrémité de la ligne et que nous les positionnons juste à l'extérieur des parois de la galerie, la ligne de section ne sera dessinée que jusqu'à la position de ces nœuds. Pour ce faire, cochez les cases situées de part et d'autre du mot "smooth" pour chaque point de la ligne. Nous devons seulement cocher la case qui montre la poignée de contrôle sur la ligne (plutôt que celle qui se trouve au-delà de l'extrémité de la ligne). Déplacez ensuite le nœud de contrôle de manière à ce qu'il soit proche du mur, mais juste à l'extérieur, comme illustré ici.

La ligne de section et la section inversée doivent maintenant ressembler à ceci :



Nous pouvons ajouter d'autres sections à la topographie en créant un nouveau scrap pour chacune d'elles et en dessinant une ligne de section et un point de section sur le scrap du plan à l'endroit où nous voulons qu'elles apparaissent. Essayez d'en ajouter quelques-unes pour mettre en pratique ce que nous avons appris dans cette leçon.

## Liens supplémentaires et informations de téléchargement

Pour que *PocketTopo* fonctionne sur certains anciens PDA *Windows mobile 5.0*, vous aurez également besoin du framework dotnet sur votre PDA, ce qui peut être difficile à trouver. C'est pourquoi nous avons placé les *DotNet\_cabs* ici : [http://www.ubss.org.uk/ Files/](http://www.ubss.org.uk/Files/)

Dézippez-les et mettez-les sur une carte SD. Choisissez le bon et exécutez-le sur votre PDA (je n'ai pas trouvé de moyen facile pour savoir lequel exécuter, mais obtenir le mauvais semble juste donner une erreur, donc je les essaie juste dans l'ordre jusqu'à ce que ça marche).

### Cave Converter

Outil de conversion du format des données du relevé. Il vous permet de convertir des données existantes dans des formats qui peuvent être importés dans *PocketTopo*. Permet également de convertir les données contenant des visées latérales d'habillage en visées LRUD (gauche, droite, haut, bas) dans des formats permettant de générer des modèles 3D utilisant LRUD.

(v20151014, publié le 14 octobre 2015) <http://wscc.darkgem.com/caveconverter>

## GMapCatcher

<http://code.google.com/p/gmapcatcher/>

Liens de téléchargement sur la gauche de cette page.

## Notepad++

Un éditeur de fichiers texte à fichiers multiples, qui peut être étendu pour exécuter *Therion* directement depuis l'éditeur afin de compiler des fichiers de projet pour produire des modèles et des cartes de cavités *Therion*.

Téléchargement : <http://notepadplusplus.org>

Pour plus de détails sur la configuration de *Therion*, voir la page wiki de *Therion* :

<http://therion.speleo.sk/wiki/doku.php/contrib:externaleditors#notepad>.

## Sumatra PDF

Ce lecteur PDF gratuit est fortement recommandé pour travailler sur des projets *Therion*. Le lecteur *Adobe Acrobat* (et la plupart des autres lecteurs de PDF sur les ordinateurs *Windows*) verrouille le fichier PDF qu'il a ouvert. Vous obtenez donc une erreur si vous essayez de générer un PDF mis à jour à partir de *Therion*, à moins de fermer le fichier PDF dans le lecteur à chaque fois avant de traiter vos fichiers de projet *Therion*.

*Sumatra PDF* vous permettra de garder le fichier PDF ouvert dans le lecteur, et chaque fois que vous le mettrez à jour en traitant vos fichiers de projet *Therion*, il rafraîchira simplement le lecteur avec le nouveau PDF. <http://www.sumatrapdfreader.org>

## Terraintool

<http://www.ubss.org.uk/terraintool/terraintool.php>

Ce programme nécessite *Java* pour fonctionner, voir les instructions sur la page web.

## TopParser

Outil qui traite directement les fichiers de données topographiques *PocketTopo* (fichiers .top) et génère des fichiers de projet *Therion* prêts à être dessinés.

(v1\_2\_3, publié le 3 août 2013)

Disponible sous forme de fichier exe *Windows* (à dézipper). Pour *Linux* et *Mac* (ou *Windows* si vous avez python), vous pouvez obtenir la source python pour l'exécuter directement (je suppose que vous savez ce que vous faites !): <https://bitbucket.org/AndrewA/topparser>

## WinMerge

(v2.14.0, publié le 2 février 2013) <http://winmerge.org>

## Référence des symboles de grottes de l'ISU

Ce document de référence montre les symboles standards internationaux de spéléologie de l'Union Internationale de Spéléologie (UIS) :

[http://www.carto.net/neumann/caving/cavesymbols/uis\\_signatures\\_english.pdf](http://www.carto.net/neumann/caving/cavesymbols/uis_signatures_english.pdf)

## Niveaux de topographie de l'ISU

Les différents niveaux de précision des topographies agréés par l'UIS sont expliqués dans ce document :

<http://scholarcommons.usf.edu/cgi/viewcontent.cgi?article=1046&context=ijs>

# **ANNEXE**

**Les évolutions de *Therion* depuis 2019**



*Therion* 5.6.0 (en cours) :

*Therion* :

- \* Support des modèles de couleurs : cmyk, rgb, niveaux de gris (voir l'option color-model layout dans le Thbook)
- \* Support de la gestion des couleurs (voir l'option de disposition des profils de couleurs)
- \* Support pour l'ombrage linéaire et radial (gradients de couleur) dans METAPOST (options withlinearshading et withradialshading)
- \* Support pour des transitions douces des couleurs de fond de page dans les cartes PDF si le fond de carte est coloré suivant l'altitude
- \* Ajout d'un commutateur de mise en page pour l'ombrage lisse <off/quick> - quick est la valeur par défaut
- \* Réimplémentation du traitement interne des motifs et des couleurs transparentes dans METAPOST.

METAPOST (changements incompatibles : l'option de dessin 'withtransparentcolor', la macro 'def\_transparent\_rgb' et les couleurs transparentes prédéfinies tr\_bg, tr\_white, tr\_black ont été supprimées ; utilisez l'option de dessin withcolor <color> withalpha <number>' à la place)

- \* Réimplémentation de la conversion de MetaPost en PDF
- \* Support du système de double numérotation dans METAPOST (ceci élimine les limites actuelles de MetaPost ; voir la section sur les limites de MetaPost) limites de METAPOST; voir le chapitre Limitations dans l'annexe du Thbook).
- \* Ajout d'une option de fichier d'initialisation tex-refs-registers <on/off> (cf. chapitres Fichiers d'initialisation et Limitations dans l'Annexe du Thbook)
- \* Support pour LuaTeX comme alternative à pdfTeX
- \* Prise en charge des chemins relatifs dans TeX (\inputrel, \savepath et \loadpicture) et dans METAPOST (inputrel).
- \* Mise à jour de la traduction en serbe
- \* Bogues corrigés :
  - symbol-hide, show, color ne fonctionne pas pour les symboles définis par l'utilisateur
  - les textes ISO8859-2 dans le code source faisaient retentir un avertissement
  - suppression de SOURCE\_DATE\_EPOCH inutile dans le Makefile de thbook
  - correction de l'instabilité numérique dans l\_overhang\_SKBB
  - échange des symboles « anastomose » & « karren »
  - pas de liste des stations de continuation des scraps [#333].
  - correction de l'inversion des flèches [#245].
  - l'aperçu des couleurs ci-dessus ne fonctionne pas [#90].
  - correction de l'exportation KML [#84]
  - p\_smartlabel échoue avec un argument image
  - autres corrections mineures
- \* Intégration de la bibliothèque {fmt}.
- \* Mise à jour de la shapelib

*loch* :

\* Outils->Sélection - affiche uniquement les topographies sélectionnées

\* bogues corrigés :

- titre manquant
- problèmes d'autorotation

infrastructure :

\* support pour les constructions CMake (merci à Matěj Plch)

\* C++14 requis

---

*Therion 5.5.7 (2021-02-06) :*

*Therion :*

\* ajout des options --generate-output-crc, --verify-output-crc

\* Erreur soulevée en cas d'utilisation d'une option de ligne de commande inconnue

\* Le numéro de version se termine par '-dev' s'il existe des modifications locales non validées.

\* optimisations :

- traitement beaucoup plus rapide des stations fixes avec des données CRS associées
- vérification plus rapide des polices optionnelles

\* bogues corrigés :

- sélection de la topographie ignorée lors de l'exportation d'un modèle 3D à partir de plans d'exposition
- correction du bug "cs local"
- correction de l'artefact du pic de reconstruction 3d de scrap
- correction de l'enregistrement de la sélection
- l'export de carte svg/xhtml sans nom de fichier crée un fichier cave.pdf

*make :*

\* meilleure détection des variantes de wx-config

\* innosetup.ini est généré dans le répertoire de construction

---

*Therion 5.5.6 (2020-12-27) :*

*Therion :*

\* correction du bogue des statistiques de la carte introduit dans la version 5.5.5

---

*Therion 5.5.5 (2020-12-22) :*

*Therion :*

- \* mise à jour de la traduction italienne [#308]
  - \* correction d'un bug avec la statistique explo/topo/carto/copyright <number>.
  - \* Correction de bogues liés à une zone d'utilisation non définie, qui devrait être utilisée dans les transformations Proj.  
dans les transformations Proj
- 

*Therion 5.5.4 (2020-12-15) :*

*Therion :*

- \* ajout de layout-statistics carto|copyright-count
- \* Utilisation de pointeurs uniques dans thdatabase [#296].
- \* ajout du support pour les lectures de gradients de 270 à 360 degrés
- \* bogues corrigés
  - scrap -walls off ne fonctionnait pas [#297].
  - Diverses corrections [#293,294,295,296,298,299,301,302,303,305].

*thbook :*

- \* support des formats de pages plus petits (configuration à la ligne 5 du fichier thbook/thbook.tex)

*loch :*

- \* bogues corrigés :
  - Correction de l'affichage de Survex 3D avec LRUD [#286].
  - Lecture de l'indicateur splay des fichiers Survex 3D [#287].

*make :*

- \* autorise -j\$N [#283]
- \* permet de passer la variable WX\_CONFIG [#284].
- \* Suppression des copies locales de getline et getopt [#290,306].
- \* petites corrections

*infrastructure*

- \* utilisation des actions Github au lieu de Travis CI
- 

*Therion 5.5.3 (2020-10-17) :*

*Therion :*

\* avertissement de dépréciation de l'unité d'angle mil/mils [#129] \* augmentation de la précision de la longueur de visée dans les exportations sql [#147].

\* corrections de bugs

- calcul erroné de la position de la station lorsque celle-ci est fixée plusieurs fois.

[#269,270]

- fuites de mémoire causant des comportements aléatoires [#278].

- La valeur de l'échelle devrait être de 1 par défaut lorsqu'elle est omise dans "calibrer" [#263].

*loch* :

\* Les visées latérales d'habillage sont rendues avec une couleur plus foncée.

\* Les visées en double sont affichées par défaut

\* Petites corrections de bogues

---

*Therion* 5.5.2 (2020-10-09) :

*Therion* :

\* ajout du chemin extend ignore -- extend ignore <station1> <station2> <station3>.

\* ajout du ratio de coupe étendue -- extend <0..100>

\* support des couleurs CMYK dans les symboles MetaPost (il ne s'agit pas encore d'un support CMYK)

\* mise à jour des définitions WKT pour certaines projections intégrées

\* Les noms des projections définies par l'EPSG/ESRI sont chargés dynamiquement à partir de la base de données Proj, si Proj.

de la base de données Proj si la version de Proj >= 6

\* les relevés non datés sont listés dans les messages *Therion*

\* Mises à jour de la documentation

\* Mise à jour de la traduction allemande [#258]

\* bogues corrigés

- utilisation du code ISO correct pour la langue tchèque ('cs' au lieu de 'cz') [#251] ;

'cz' est toujours autorisé dans la mise en page de la carte pour une compatibilité descendante.

- pdftex inclus dans le programme d'installation de Windows ne trouvait pas les images avec avec './.' ou './..' dans le chemin du fichier.

- traitement des plans d'affichage de la coupe développée

- autre correction pour l'échange d'axes de Proj 7.1.0 dans la projection de Krovak

- refuse de compiler avec des versions de Proj non supportées

- la profondeur est incorrectement prise à partir des jambes de la surface

- correction de l'erreur de lecture de Survex

- Corrections de la compatibilité avec VTK9

- correction du géoréférencement de la carte *XTherion*

- corrections de compatibilité mingw-w64

- problèmes de compilation de wxWidgets
- déclinaison fixée à 0 en dehors de la plage du modèle geomag (maintenant la déclinaison est extrapolée 15 ans après la date du modèle ; l'erreur est produite après cela et avant 1900)
- autres corrections mineures

infrastructure :

- \* ajout de plus de cibles Travis CI, mise à jour des bibliothèques utilisées
  - \* suppression de la reconstruction de la bibliothèque *Therion*
- 

*Therion* 5.5.1 (2020-07-03) :

*Therion* :

- \* suppression de la longueur de visée en double dans les statistiques individuelles des membres de l'équipe.
- \* amélioration du support de Proj en mode automatique (spécification automatique de la zone d'utilisation ; mise en cache et journalisation des transformations utilisées)
- \* Ajout d'un support pour la gestion des grilles de transformation de coordonnées manquantes.  
(voir l'option proj-missing-grid init file)
- \* support de la liaison avec de multiples versions de Proj pour améliorer les tests.  
(voir test\_proj.sh)
- \* Amélioration de la compatibilité avec mingw-w64
- \* bogues corrigés
  - Diverses corrections liées à la mémoire par Matěj Plch [#242,243,244,248].
  - construction de débogage cassée [#246]
  - contournement de l'échange d'axes de Proj 7.1.0 dans la projection de Krovak
  - Divers éléments manquants dans les cibles propres [#252].

Installateur windows :

- \* mise à jour de InnoSetup
  - \* mise à jour de la distribution de TeX et METAPOST
  - \* l'installateur inclut maintenant automatiquement la bonne version des fichiers de support de Proj
- 

*Therion* 5.5.0 (2020-05-01) :

*Therion* :

- \* fonctionnalité maps-offset <on|off> dans le fichier *thconfig* pour désactiver les cartes dessinées en décalage [#159]
- \* fonctionnalité maps-offset <on|off> dans le fichier *thconfig* pour désactiver les cartes dessinées en ignorant la définition des cartes

- \* Fonctionnalité log extend dans *thconfig* pour enregistrer les constructions de coupes développées.
- \* ajout du point « mudcrack » (merci à Rodrigo Severo)
- \* Ajout d'un jeu de symboles SBE étendu (merci à Rodrigo Severo)
- \* mise à jour des données geomag jusqu'en 2025
- \* Ajout du support pour la génération reproductible des sorties PDF et SVG.  
(option de ligne de commande --reproducible-output)
- \* rendre reproductible la construction de thbook.pdf (dériver les dates de création/modification du fichier PDF à partir de la date de livraison)
- \* Amélioration du support de Proj 6.0 et 7.0 (voir l'option proj-auto init file).
- \* Ajout de la bibliothèque de tests unitaires Catch2 et de cas de tests Proj.
- \* tous les scripts python utilisent maintenant Python 3
- \* Ajout des traductions serbe (merci à Ivana Miskovic) et slovène [PR#142].
- \* Mise à jour de la traduction portugaise [PR#170,220].
- \* Améliorations du thbook par Benedikt Hallinger [PR#161,162].
- \* bogues corrigés
  - orthographe dans certains chapitres du thbook
  - sortie html et kml [PR#145,150].
  - Correction de l'extension ingore <station>.
  - Il manque à la sortie de Survex 3D les noeuds à la fin des visées d'habillage anonymes [#157]
  - a\_blocks\_AUT manque le point-virgule [#126]

#### *XTherion* :

- \* Ajouter *thconfig*\* à la liste des fichiers de configuration sélectionnables [PR#168].
- \* bogues corrigés :
  - Correction de la géométrie des fenêtres de Windows *XTherion*.

#### *loch* :

- \* bogues corrigés :
  - compilation MacOS X [PR#144].
  - plusieurs corrections mineures
  - Correction d'un bogue dans le rendu hors écran de Linux

---

#### *Therion* 5.4.4 (2019-05-01) :

#### *Therion* :

- \* support de l'API Proj 5.0 et plus récente (proj.h)
- \* bogues corrigés :
  - erreur de segmentation lors de la génération de SVG lorsque compilé par un g++ plus récent
  - gestion incorrecte des projections EPSG si plus d'un code EPSG est utilisé

- nom de projection personnalisé manquant
  - noms de projection lisibles par l'homme manquants dans Proj4
- 

*Therion* 5.4.3 (2019-02-01) :

*Therion* :

\* longueur de la ligne d'entrée augmentée pour permettre la lecture de jeux de données de coupe plus importants.

\* Suppression de la projection Proj.4 intégrée ; remplacement de la correspondance interne des codes EPSG et ESRI

par +init=epsg|esri :... chaîne d'initialisation libproj

\* bogues corrigés :

- ajouter et utiliser les fichiers de grille PROJ dans la distribution Windows
  - correction de commentaires et de fautes de frappe dans le code [PR#127].
-