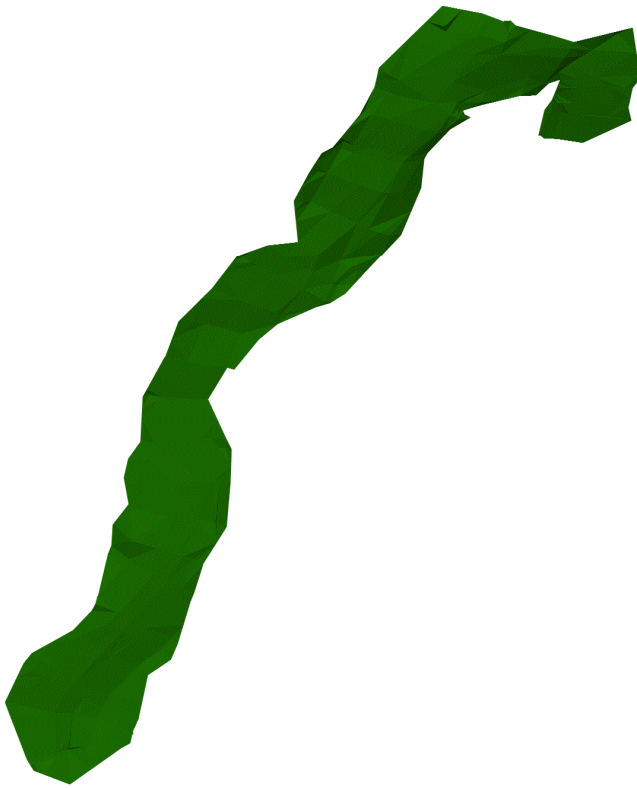


2012-
2013

Rapport de projet



Nicolas BIANCHI

Licence Pro CAO DAO

Tuteur : Jean-Marc PYRÉ

Professeur référent : Jean-Claude SCHUBER



Création d'un programme LISP permettant l'insertion d'une
polyligne en 3D grâce au profil en long et aux profils en
travers d'une piste.

Remerciements

Je tiens tout d'abord à remercier Monsieur Louis GUILY, président directeur général, pour avoir accepté de m'accueillir au sein de sa société afin de réaliser ma licence professionnelle CAO/DAO.

Je souhaite également remercier Monsieur Jean-Marc PYRÉ, mon tuteur, pour m'avoir également permis de réaliser mon année en alternance au sein de l'entreprise et pour son encadrement au cours de cette année.

Je souhaite énormément remercier Monsieur Julien MORENO, qui a su me guider dans mes premiers pas dans le métier de dessinateur-projeteur, pour sa collaboration, ses conseils avisés et surtout pour le temps qu'il m'a consacré durant cette année.

Je tiens à remercier mes collègues pour l'accueil qu'ils m'ont réservé et pour leur patience car à leurs côtés j'ai pu progresser, apprendre et profiter de leurs expériences respectives.

Cette première expérience sera toujours très importante pour ma carrière. Les tâches auxquelles j'ai été associé m'ont vraiment permis de consolider mes connaissances et d'en développer de nouvelles.

SOMMAIRE

Remerciements	1
Avant-propos	4
Introduction	5
1. L'entreprise	6
1.1. Situation géographique	6
1.2. Activités pratiquées	7
1.3. Principaux consultants mobilisés au sein de DIANEIGE :	8
1.4. L'organisation de l'entreprise	10
1.5. Quelques réalisations en images	12
1.6. Projet réalisé durant l'année	14
2. Le Projet	17
2.1. Rappels concernant la vue en plan, le profil en long et le profil en travers	17
2.2. Le Projet	19
2.3. Fin du programme	20
3. Ecriture du LISP	26
3.1. Premiers essais	26
3.2. Évolution du programme	30
4. Problèmes rencontrés	43
4.1. Récupération d'attributs dans la routine Routine OrientationEtCoordonnéesBloc et ExtractionPDC	43
4.2. Orientation du PT dans la routine Routine OrientationEtCoordonnéesBloc	43
4.3. Stockage des sommets de la polyligne 2D dans la routine importation	43
4.4. Sélection calque gelé dans la boîte de dialogue	44
4.5. Rendu visuel de la boîte de dialogue	44
5. Conclusion	45
6. Lexique	46
7. Annexes	47



ENTREPRISE :

DIANEIGE

16i Chemin de Malacher

38240 Meylan



INSTITUT UNIVERSITAIRE DE TECHNOLOGIE :

IUT de Nîmes

CS 12007

8 rue Jules Raimu

30907 Nîmes Cedex 2

Avant-propos

Dans le cadre de notre formation en licence professionnelle CAO-DAO multimédia dans le bâtiment et les travaux publics à l'IUT de Nîmes, nous devons réaliser un projet de fin d'études, afin de valider notre année en alternance.

Ce projet doit nous permettre de mieux nous intégrer dans l'entreprise et de mettre en avant nos compétences acquises au sein de l'IUT au profit de celle-ci. Ce projet doit également permettre une augmentation de compétences à l'entreprise.

Introduction

La licence CAO DAO est un passage déterminant pour mon avenir. Les objectifs de cette année en alternance, réalisée après mon DUT Génie Civil, étaient d'apprendre à utiliser de nouveaux logiciels de travail, de perfectionner mes acquis, d'acquérir une expérience professionnelle, ainsi que conforter mes précédents choix scolaires et mon orientation professionnelle.

Au cours de cette année, que ce soit à l'IUT ou encore au sein de l'entreprise, tout me fit penser que je suivais la voie qui me correspondait.

En effet, la générosité et la patience dont on fait preuve mes collègues m'ont permis de découvrir le métier de projeteur sous toutes ses facettes ainsi que le fonctionnement de l'entreprise. De plus, les responsabilités qui m'ont été confiées, m'ont révélé les difficultés du métier.

Quant à l'IUT, ses exigences m'ont permis de découvrir et de maîtriser de nouveaux logiciels incontournables dans le domaine du BTP, ainsi que consolider mes acquis sur les logiciels que j'utilisais déjà ; Les cours abordés à l'IUT m'ont permis de mieux appréhender mes capacités, mes difficultés, et, d'affiner mes préférences.

Aussi bien que mes périodes en entreprise que celles d'enseignement, m'ont été bénéfiques professionnellement tout au long de cette année et ont confirmé encore une fois mes choix professionnels.

Durant ces alternances, nous avons un projet de fin d'études à réaliser, marquant la touche finale du cursus.

Ce projet, consistant à concevoir un programme, a permis d'apporter un peu de notre savoir faire au bénéfice de notre entreprise d'accueil. Dans le cas de mon projet, le bénéfice sera plutôt temporel. En effet, le projet qui m'a été confié, est la réalisation d'un programme simple d'utilisation : dessiner une polygone en trois dimensions, représentant le projet d'aménager une piste de ski.

Ce rapport, quant à lui, va me permettre de faire partager un peu de mon quotidien durant l'année.

Pour que ce «bilan» soit le plus objectif et le plus intéressant possible, je vais vous faire part de mes erreurs, de mes difficultés et de mes doutes aussi bien que de mes réussites. Il s'agit donc, tout au long de cette lecture, de suivre pas à pas mes découvertes dans l'entreprise ainsi que ma réflexion concernant la création de mon projet.

Au fil de ce rapport, je vais donc vous présenter le bureau d'études et particulièrement le pôle neige dans lequel j'ai eu la chance d'évoluer. Je consacrerai donc, une seconde partie, à la justification et à l'explication de mon projet de fin d'études. De plus, je vous expliquerai également par quels moyens je suis parvenu à réaliser le programme qui m'a été confié.

1. L'entreprise

Au cours de mon année de licence professionnelle CAO/DAO, j'ai réalisé mon alternance dans le bureau d'études DIANEIGE présentée dans cette première partie.

1.1. Situation géographique

Dianeige est un bureau d'études (SA), créé en 1990 à Courchevel par Monsieur Louis Guily maintenant situé dans le département de l'Isère à Meylan, spécialisé dans l'aménagement touristique de la montagne.

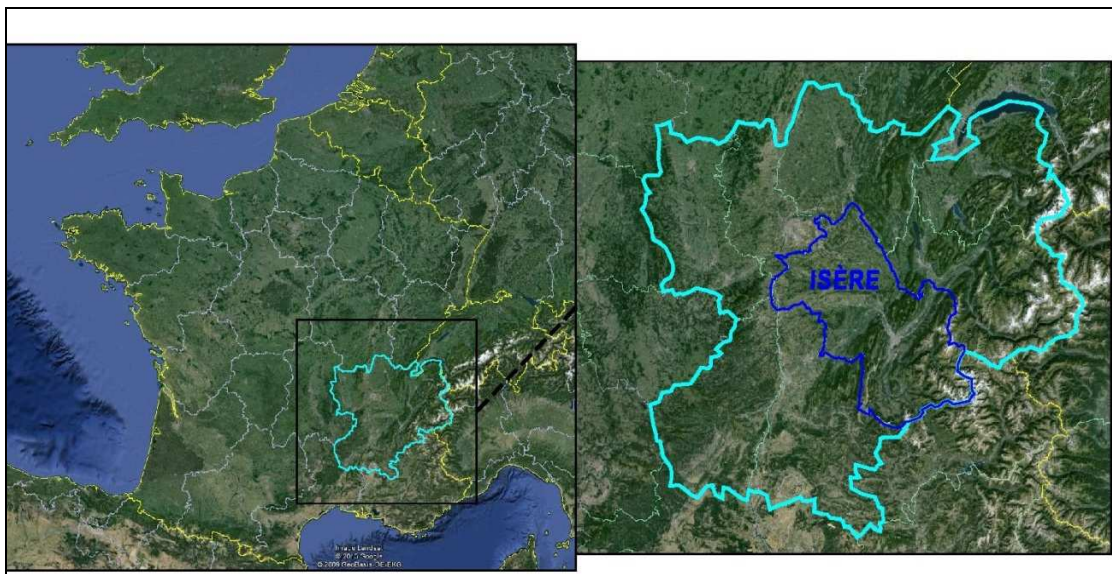


Figure 1 : Localisation départementale

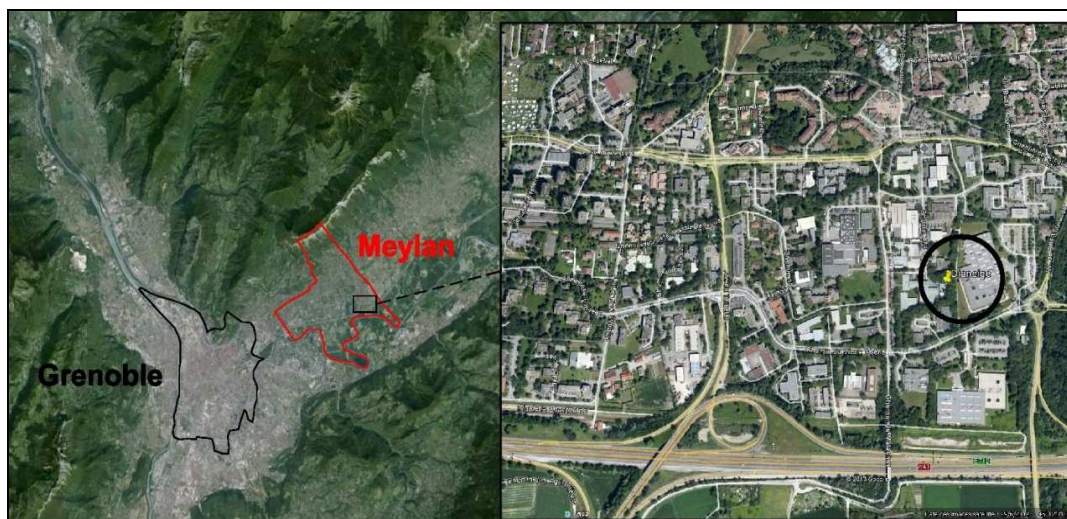


Figure 2 : Localisation communale

1.2. Activités pratiquées

Dianeige est spécialisée dans l'aménagement touristique de la montagne et intègre des compétences allant du marketing touristique à la maîtrise d'œuvre, ce qui lui confère une approche globale du développement touristique des sites de montagne.

- Aménagement de stations touristiques de montagne ;
- Spécialiste des aménagements dédiés à la pratique du ski et des activités de neige ;
- Maîtrise d'œuvre de projets d'aménagement.

Ses compétences s'étendent dans les domaines du tourisme, de la communication et du développement local. DIANEIGE développe depuis de nombreuses années une expertise du schéma directeur d'aménagement de domaines skiables en France (Alpes, Pyrénées, Jura, Vosges) et à l'étranger (Roumanie, Bulgarie, Japon, Russie...) aussi bien en création qu'en restructuration.

Ses compétences et expériences permettent au bureau d'études Dianeige de justifier les qualifications **OPQIBI** (Organisme Professionnel de Qualification de l'Ingénierie Bâtiment Industrie) suivantes :

- Etudes loisirs, culture et tourisme (0501) ;
- Terrassements – Voiries – Réseaux enterrés (étude en terrassements complexes) (1102) ;
- Ingénierie des ouvrages et systèmes d'infrastructure (1803).

Ci-dessous un extrait du cahier de références de DIANEIGE :

- Aménagement et restructuration des domaines de Gresse en Vercors, Col de l'Arzelier, Col d'Ornon et Les Signaroux (38) ;
- Définition d'un nouveau positionnement de la station de Luz Ardiden (64) basé sur des activités diversifiées (2012) ;
- Schéma directeur d'aménagement et de restructuration de la station de La Féclaz (73) ;
- Elaboration de scénarii de diversification des activités touristiques dans les sites de moyenne montagne pour le compte du Conseil Général de l'Isère ;
- Etude de repositionnement et redéploiement de la station de Val Louron (65) dans l'objectif d'une valorisation touristique « quatre saisons » ;
- Plan de développement multi-saisons du plateau de Payolle ;
- Étude pour la mise en cohérence de l'offre neige du Massif de Chartreuse ;
- Schéma de gestion de la station de Cauterets (2004 mise à jour en 2009) prévoyant la restructuration à long terme du domaine, implantation du télésiège débrayable des Crêtes (2005), Avant-projet et permis avec passage à la commission des sites pour la piste des Crêtes, terrassement de 125 000 m³ (2009), permis pour l'aménagement de pistes de VTT en 2008 ;
- Schéma directeur d'aménagement de la station de Peyragudes : mise en œuvre du deuxième schéma d'aménagement à 10 ans : aménagement de l'espace débutants, liaison urbanisation de Balestas et front de neige, restructuration du secteur de Serre Doumenge, Boarder cross... ;
- Schémas directeurs de création ou de restructuration de nombreux sites à l'étranger : Perelik (Bulgarie), Xiling (Chine), Cumpana (Roumanie), Arkys (Russie), Ordino Arcalis (Andorre)...

A ce jour, ce sont plus de mille pistes conçues, deux millions de mètre cube d'eau stockée, plus d'un million de mètre cube d'eau turbinée pour les installations de neige de culture, des centaines de dameurs, pisteurs-secouristes et nivoculteurs (ou snowmakers ; Ils gèrent le réseau de neige de culture, le bon fonctionnement de l'installation et sont donc responsables de la qualité de la neige) qui ont suivi les formations de Dianeige sur la préparation des pistes de ski et l'optimisation du service damage.

Dianeige a mis en place une démarche originale « HQE Aménagement » liée à notre volonté de respecter selon diverses valeurs le patrimoine naturel. Nos projets d'aménagement de territoire touristique en montagne ont par conséquent un point commun : le respect de l'environnement.

1.3. Principaux consultants mobilisés au sein de DIANEIGE :

Louis Guily : PDG et fondateur de Dianeige - 25 ans d'expérience.

Titulaire d'un Doctorat en nivologie ; géographe, aménageur, il sera force de propositions au moment de la définition des axes stratégiques et des scénarii d'aménagement et apportera son expertise pour évaluer de la cohérence globale du projet. Il assurera le suivi du bon déroulement de la mission vis-à-vis des objectifs de qualité du bureau d'études Dianeige.

Alain CHOMEL : Chargé d'affaires

Titulaire d'un DESS Aménagement et développement de la montagne, il dispose à son actif de plusieurs missions en milieu de montagne. Depuis 1994, il développe et anime plusieurs missions autour des :

- stratégies de diversification multi-saisons neige – hors neige (repositionnement station – « resort ») ;
- marketing touristique et marketing produit lié aux loisirs,
- définition et faisabilité d'équipements de loisirs ou culturels, animation/gestion de projet...).

Alain CHOMEL sera l'un des principaux intervenants sur la mission.

Nicolas Brunel : chargés d'affaires Dianeige, 14 ans d'expérience,

Il est titulaire d'un diplôme d'ingénieur maître « Génie de l'environnement option aménagement » d'un DESS « aménagement, gestion et équipement des pays de montagne ». Il a acquis quatorze années d'expérience en matière d'aménagements touristiques. Ses interventions au sein de Dianeige sont plus spécifiquement celles liées à la réalisation de schémas directeurs d'aménagement de domaine skiable, aux pistes de ski et aux projets d'activités de loisirs de pleine nature (intervention sur les études de définition et les études opérationnelles).

Monsieur Brunel sera l'interlocuteur privilégié du Maître d'Ouvrage, il assurera le pilotage global de l'intervention des différents consultants et conduira les missions d'études relatives aux potentiels de développement de l'offre de loisirs, aux projets d'aménagement et à l'évaluation des budgets d'investissement.

Olivier Mermillod: DGD de Dianeige.

Titulaire d'un diplôme d'ingénieur Arts et Métiers. Il apportera son expérience d'exploitant de domaine skiable et sa compétence en matière d'audit. Ces interventions concerneront pour l'essentiel les problématiques de l'exploitation, de la remontée mécanique et de l'économie de projet.

Jean Marc Pyr  : Charg  d'affaires.

Ing nieur g otechnicien ISTG : sp cialis  en  tudes techniques et ma trise d' uvre pour ce qui concerne la conception de pistes de ski, de r seaux de neige de culture et r serves d'altitude. Jean Marc Pyr  apportera son exp rience des travaux en montagne pour tous les aspects techniques de la conception.

Eric Artiga : Charg s d'affaires.

Licenci  en Arts Plastiques (Paris I – Sorbonne), Dipl m  d'un Master (DPEA) en design pour l'architecture (Ecole Nationale Sup rieure d'Architecture de Grenoble), Titulaire d'un Master UHCI (Urbanisme Habitat et Coop ration Internationale) de l'IUG (Institut d'Urbanisme de Grenoble), Eric Artiga apportera sa contribution sur les probl matiques li es   la valorisation du patrimoine naturel, culturel ou paysager,   la signal tique et au design des projets.

1.4. L'organisation de l'entreprise

Comme je l'ai dit précédemment, j'ai passé mon année d'alternance au sein du bureau d'étude DIANEIGE.

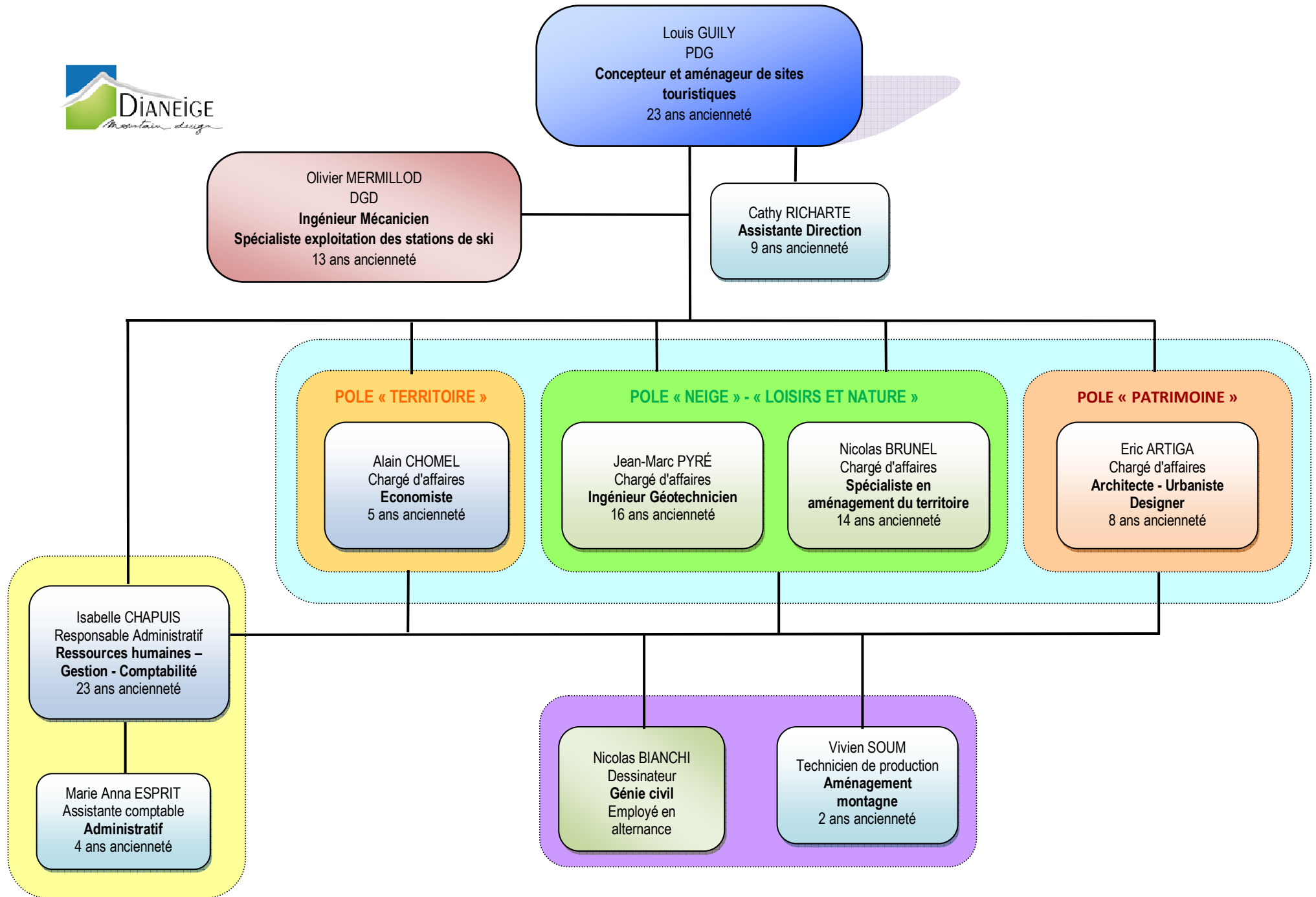
Il possède également deux filiales :

- KALIBLUE : Agence de communication spécialisée dans la cartographie et l'imagerie 3D artistique ;
- DIATHEMIS : Conseil pour les affaires et autres conseils de gestion



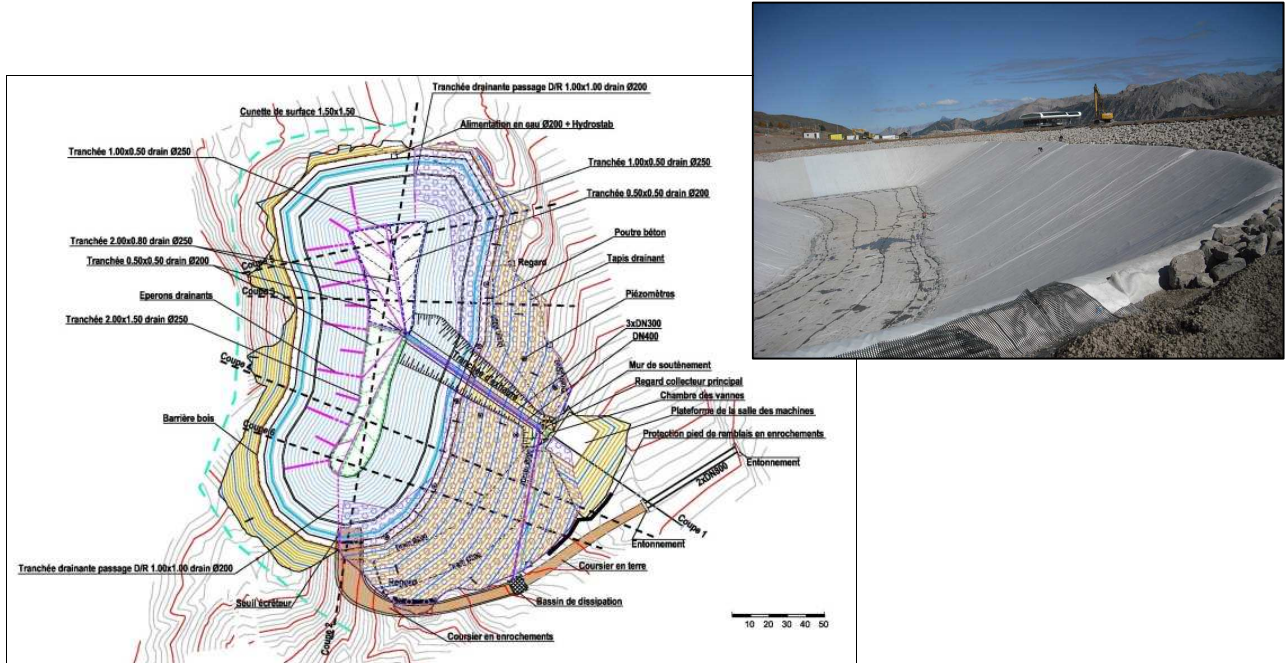
Figure 3 : Fonctionnement du bureau d'études

Le bureau d'études compte dix employés (voir ci-après)

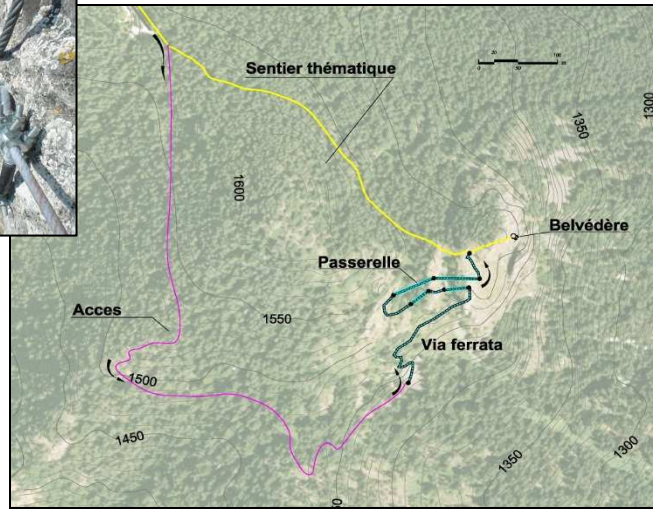


1.5. Quelques réalisations en images

MAITRISE D'ŒUVRE COMPLETE – TRAVAUX D'AMENAGEMENT
DE LA RETENUE D'ALTITUDE DE 125.000 M³ DE LA STATION DE VARS (05)



**ETUDES DE PROJET POUR L'AMENAGEMENT DES PARCOURS DE VIA FERRATA
SITE DE MONTAGNE DE PERELIK (BULGARIE)**



**MAITRISE D'ŒUVRE COMPLETE - TRAVAUX D'AMENAGEMENT
PISTES DE SKI STATION DES DEUX ALPES (38)**



1.6. Projet réalisé durant l'année

Au cours de cette année en alternance, j'ai eu la chance de travailler sur de nombreux projets et ainsi, produire une multitude de documents différents. Je vais à présent vous exposer un projet, que j'ai trouvé particulièrement intéressant et complet, concernant la station de Vars dans les Hautes-Alpes.

A la demande et pour le compte de la **SemSedev** (Société pour l'Équipement et le DEveloppement des remontées mécaniques et des pistes de Vars), opérateur du domaine skiable, Dianeige a réalisé une étude d'aménagement de pistes de ski sur le domaine skiable de **Vars (05)**. La station de Vars, reliée à celle de RISOUL, constitue l'un des plus vastes domaines skiables des Alpes du Sud : "La Forêt Blanche".

Ces agencements sont en relation avec la création du télésiège débrayable 6 places (TSD6) du Lièvre remplaçant le Télési (TK) du même nom et le télésiège fixe (TSF) de la Mayt.

La réalisation du permis d'aménager de ce projet m'a été confiée.

Les aménagements concernent les pistes suivantes :

- La plateforme de débarquement du TSD6 du Lièvre.
- La restructuration de la piste bleue « Gentianes »
- La création d'une variante rouge « Gentianes »
- La restructuration de la piste des Sagnes.

Ces études sont menées au stade de Projet. Elles s'appuient sur :

- Le cahier des charges défini avec la SemSedev.
- La visite de terrain du 29/08/2012.
- Le relevé terrestre du site au 1/1 000^{ème} réalisé par Kalibblue – Relevé GPS dossier Ka803 - Juillet 2012.
- Le fond réalisé par photogrammétrie au 1/4 000^{ème} édité par Kalibblue - Dossier Ka634 - Janvier 2012.
- Les données CNA concernant la gare amont du TSD6 du Lièvre - Mail du 23 août 2012.
- L'étude hydraulique du RTM concernant le torrent des Claux – Juillet 2011.
- La thèse de Brice Martin concernant les relations entre facteurs naturels, facteurs anthropiques et aléas à Vars – 1996.

L'extrait de carte IGN ci-dessous permet de localiser la station dans son environnement régional.

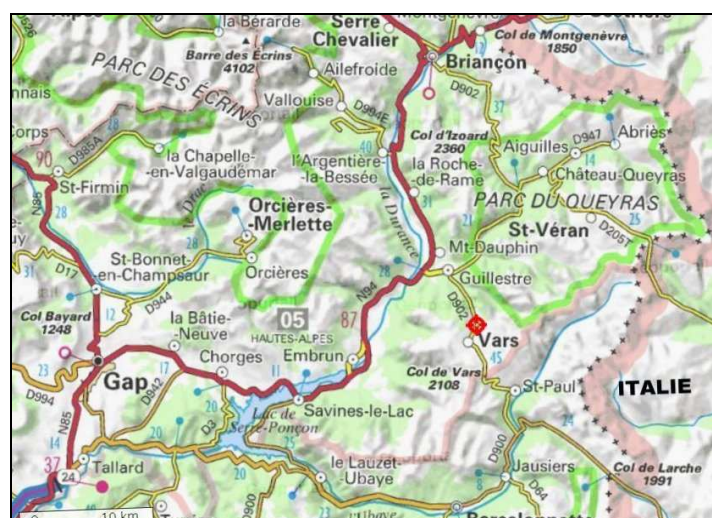


Figure 4 : Extrait de la carte IGN au 1/25000^{ème}

Le plan suivant localise au sein de la station les secteurs aménagés dans le schéma de pistes selon :

- En bleu = station de Vars.
- En rouge = station de Risoul.
- Pointillés = remontées mécaniques démontées.

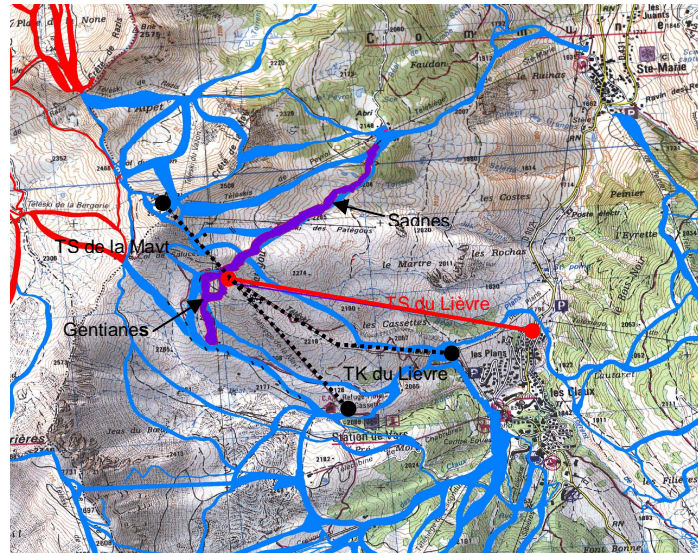


Figure 5 : Localisation des ouvrages IGN



Figure 6 : Localisation ortho photographie

Les aménagements prévus suivent :

- La restructuration de la piste Gentianes pour l'afficher un niveau bleu ;
- La création d'une variante plus raide, en sortie gauche du télésiège ;
- La restructuration de la piste Sagnes en niveau bleu permettant l'accès au secteur de Peyrol.

Les pièces que j'ai réalisées sont :

Licence Professionnelle CAO/DAO Dessinateur Projeteur

Année 2012/2013

- PA1 - Plan de situation du terrain 1 / 2 000^{ème} (Annexe 1)
- PA2 – Notice décrivant le terrain et le projet d'aménagement prévu (Annexe 2)
- PA3 - Plan de l'état actuel du terrain à aménager et de ses abords 1/1 500^{ème} (Annexe 3)
- PA4 - Plan de composition d'ensemble du projet coté dans les trois dimensions 1/1 500^{ème} (Annexe 4)
- Plan de situation (Annexe 6)
- Plan cadastral déboisement (Annexe 7)
- Plan reboisement (Annexe 6)

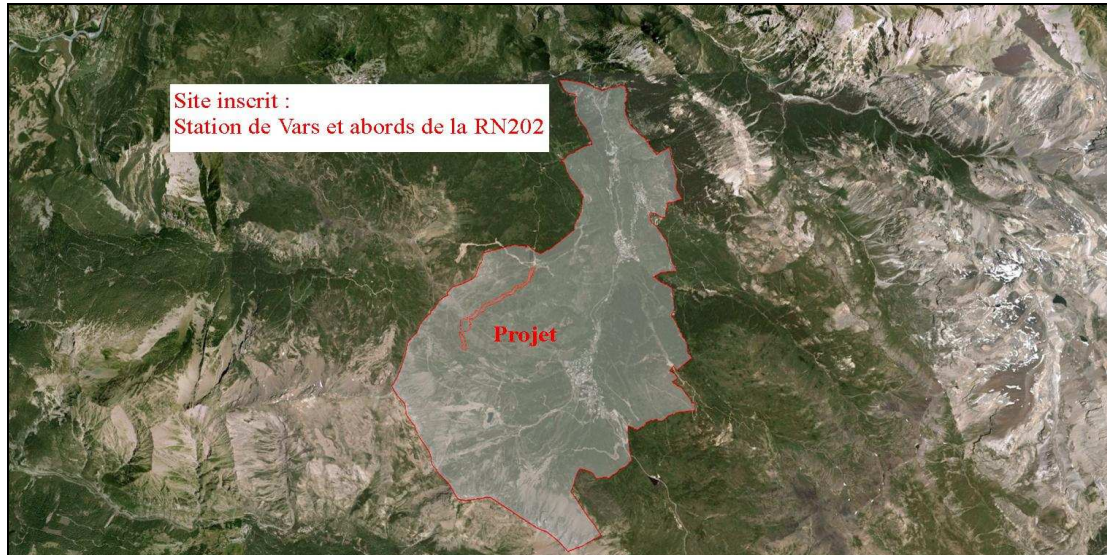


Figure 7 : Localisation dans le site inscrit

Le projet se situe dans le site inscrit «Station de Vars et abords de la RN202». Les parcelles boisées concernées par le défrichement sont soumises au régime forestier, c'est-à-dire une gestion des forêts sous la responsabilité de l'ONF. Un dossier de demande de défrichement est déposé en parallèle du permis d'aménager.

2. Le Projet

2.1. Rappels concernant la vue en plan, le profil en long et le profil en travers

⇒ **Vue en plan**

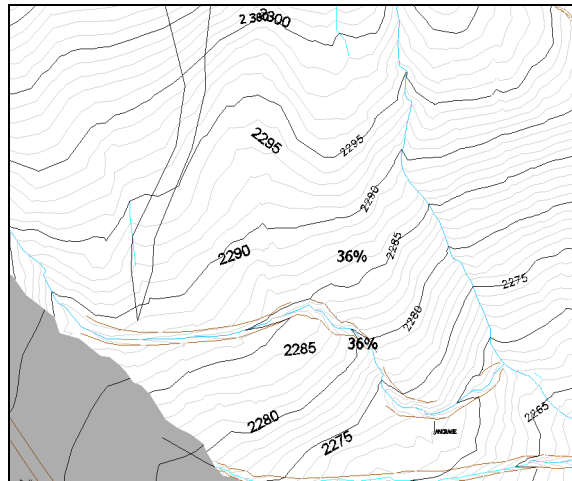


Figure 8 : Vue en plan piste avant projet

Sur la vue en plan les courbes de niveau sont destinées à donner un aperçu du relief réel. Une courbe de niveau est l'intersection du relief réel avec un plan horizontal d'altitude donnée en cote ronde (généralement un nombre entier). Les courbes sont équidistantes en altitude ; leur espacement horizontal dépend de la déclivité du terrain à représenter et de l'échelle du plan ou de la carte.

Lors d'un avant-projet sommaire de l'étude d'un projet, le projeteur a besoin d'une vue en coupe du terrain naturel suivant l'axe du projet qu'il étudie : ce graphique est le profil en long du terrain naturel. Quelques vues en coupe perpendiculaire à l'axe sont aussi nécessaires à l'étude : ce sont les profils en travers. Ces deux types de graphiques permettent d'obtenir, après plusieurs études un tracé « idéal » répondant aux impératifs du projet que sont l'emprise sur le terrain, les pentes maximales, le moindre coût, etc.

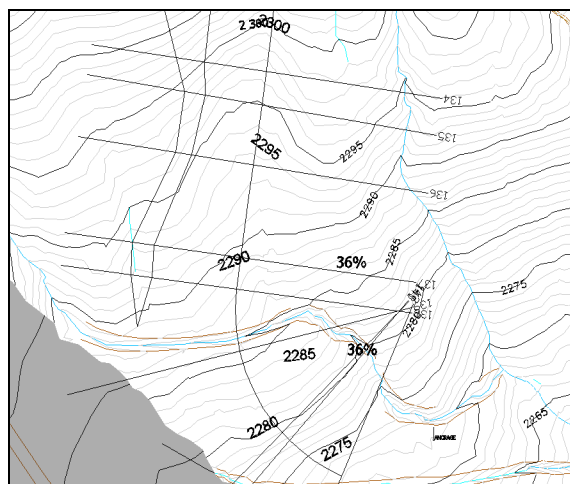


Figure 9 : Vue en plan avec axes profils en long et en travers

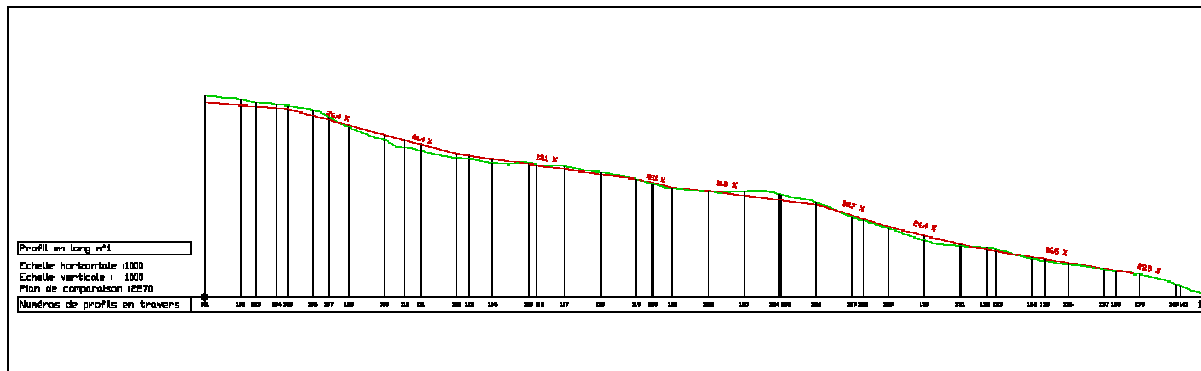


Figure 10 : Profil en long

Le profil en long est un graphique sur lequel sont reportés tous les points du terrain naturel et de l'axe du projet. Il est établi en premier lieu. (*L'échelle de représentation est de l'ordre de 1/2000 à 1/500*). On s'appuie sur ce document pour le dessin des profils en travers. Distances et altitudes sont données en mètres au centimètre près. On choisit en général un plan de comparaison d'altitude inférieure à l'altitude du point le plus bas du projet ou du terrain naturel. Ce plan de comparaison est l'axe des abscisses du graphique sur lequel sont reportées les distances horizontales suivant l'axe du projet. Sur l'axe des ordonnées, sont reportées les altitudes.

On dessine tout d'abord le terrain naturel (TN), généralement en trait moyen vert. Son tracé est donné par la position de chaque point d'axe d'un profil en travers, le terrain naturel étant supposé rectiligne entre ces points. Les calculs des positions des points caractéristiques se ramènent à des intersections droites-droites, droites-cercles ou droites-paraboles dans le repère associé au profil en long.

⇒ Profil en travers

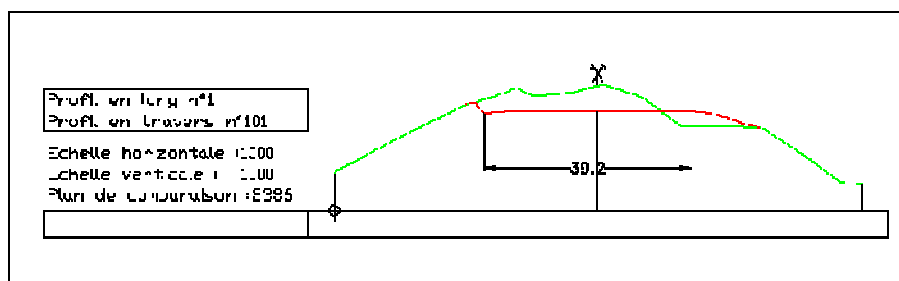


Figure 11 : Profil en travers

Les profils en travers (*sections transversales perpendiculaires à l'axe du projet*) permettent de calculer les paramètres suivants :

- la position des points théoriques d'entrée en terre des terrassements ;
- l'assiette du projet et son emprise sur le terrain naturel ;
- les cubatures (volumes de déblais et de remblais).

Le profil en travers est représenté en vue de face pour une personne qui se déplacerait sur l'axe du projet de l'origine à l'extrémité du projet (pour une piste de haut en bas et pour une remontée mécanique de bas en haut). On commence par dessiner le terrain naturel à partir d'un plan horizontal de référence qui n'est pas forcément celui du profil en long, de manière à obtenir le profil en travers à l'échelle maximale sur le format choisi. (*L'échelle de représentation est de l'ordre de 1/500 à 1/100*). Il n'y a pas d'échelle différente en abscisse et en ordonnée de manière à pouvoir mesurer directement sur le graphique des longueurs dans toutes les directions ou bien des surfaces. L'abscisse de chaque point du terrain naturel (ou du projet) est repérée par rapport à l'axe du profil en travers (*donc négative à gauche et positive à droite*), l'ordonnée est toujours l'altitude du point. On y superpose ensuite le projet à partir du point d'axe dont l'altitude a été déterminée sur le profil en long. Cela permet de calculer la position des points d'entrée en terre. Il existe trois types de profils en travers : les profils en remblai, en déblai ou bien les profils mixtes.

La projection de la piste sur le plan horizontal (*sur la vue en plan*) se représente sur le plan topographique (*support au tracé ; échelle : 1/500^{ème} à 1/100^{ème}*). On y représente l'axe de la route (profil en long) et on y reporte la position des profils en travers.

2.2. Le Projet

Le but de mon projet est donc de réaliser la projection de la piste non pas sur un plan horizontal mais en trois dimensions et dans le but de faciliter le travail des projeteurs du bureau, de réduire leur temps de conception et de modélisation de pistes de ski en 3D tout particulièrement. La conception de pistes est une tâche qui demande de la précision (*implantation, calculs de volume de déblais et remblais*)

C'est pourquoi, on m'a demandé de créer un programme, à partir d'un projet d'aménagement de pistes, permettant la création et modélisation de polygones en trois dimensions représentant la piste projet (*piste à créer ou à modifier*). Les différents dessins que l'on peut trouver dans le projet sont la vue en plan du site, les profils en long d'une ou plusieurs pistes et ceux en travers de ces mêmes pistes. Les informations contenues sur la vue en plan et les profils en travers me permettront de modéliser la piste.

Le programme permet la création d'un Model Numérique de Terrain (MNT) projet en 3D pour ensuite pouvoir réaliser les calculs de volume de terrassement (*cubatures*) en comparant le MNT projet à celui du terrain naturel. Il permettra aussi de mesurer plus précisément les mètres du projet : surface, distance, pentes. Cela donne lieu aux calculs des prix projets, de faire des projets réalistes et permet de vérifier les problèmes liés aux raccordements entre les profils en travers et les entrées en terre. Des images d'intégrations 3D pourront également être réalisées de ce fait.

Les profils en travers du programme utilisé sont réalisés à toutes les étapes du projet. A savoir qu'une esquisse ne produit qu'un ou deux profils en travers, en avant projet un tout les cinquante mètres et en projet un tout les vingt mètres.

Mon travail sur ce programme sera essentiellement sur Auto Lisp, langage de programmation intégré à AutoCAD dérivé du LISP. Le terme LISP est un acronyme (*LISt Processing*) signifiant "Traitement de liste". Le LISP est un langage interactif et compilé, conçu dans les années 1950 par John Mac CARTHY.

2.3. Fin du programme

Une fois le programme terminé, à l'aide de Covadis (ou d'un autre programme Lisp que j'ai eu la chance de pouvoir récupérer), il est possible de réaliser le modèle numérique de terrain (MNT) du projet de la piste créée.

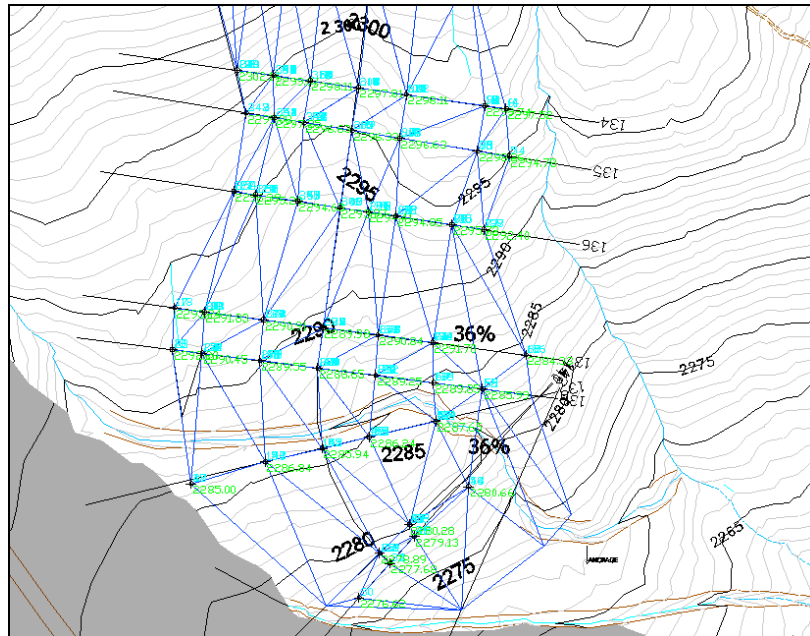


Figure 12 : Points et polygones du programme + MNT

Les points forment les sommets des faces 3D et les polygones 3D peuvent être utilisés comme ligne de rupture.

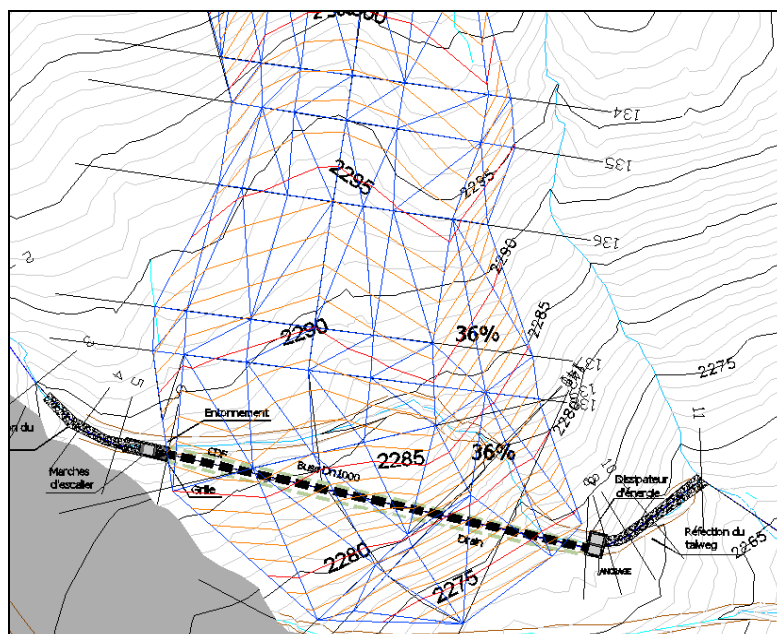


Figure 13 : Modèle Numérique de Terrain + Ligne de rupture projet

Une fois réalisé, nous pouvons réaliser les calculs projet lignes de rupture, volumes de déblais/remblais, entrées en terre, pentes du projet et l'estimation financière.

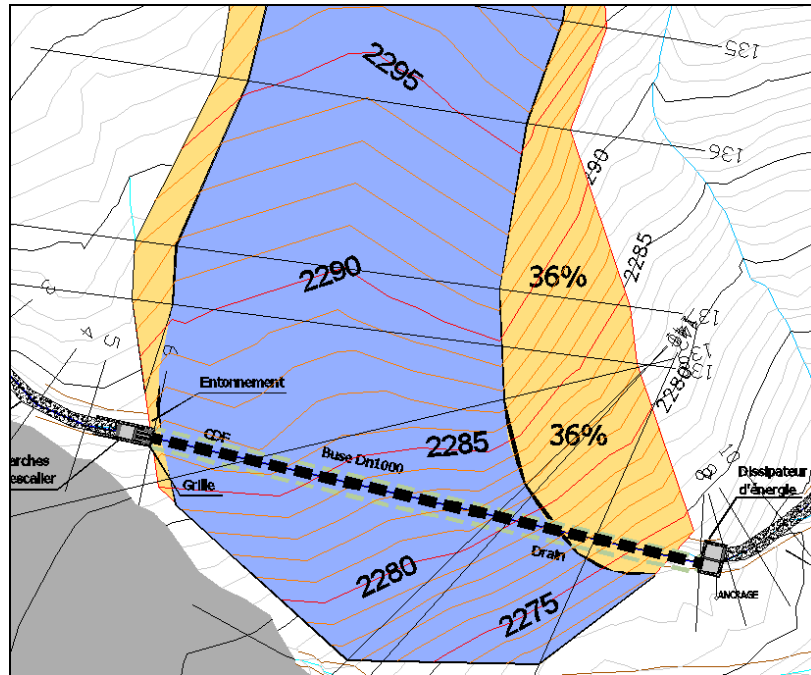


Figure 14 : Implantation piste projet

Le projet nous permet d'apercevoir la piste en trois dimensions comme ci-dessous :

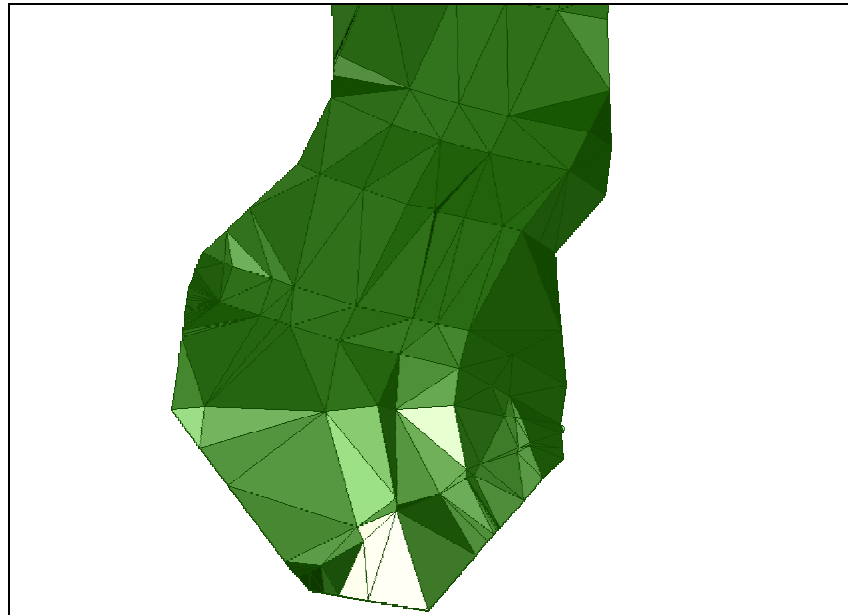


Figure 5 : Rendu du MNT

Création du programme

Pour commencer, j'ai analysé le projet qui m'a été confié, de façon à cerner les besoins du programme, et, dans le but de trouver les points importants pour commencer la rédaction du programme.

Ce projet (Annexe 10) est à utiliser après avoir réalisé, grâce à Covadis⁽¹⁾, les coupes de profils en travers de la piste étudiée car je me sers des informations contenues dans les blocs⁽²⁾ créé par Covadis. Toutefois, j'ai également conçu un programme permettant de créer les polygones⁽³⁾ 3D souhaitées, sans avoir à passer par les blocs créés par Covadis lors des coupes. La programmation est beaucoup plus lourde à utiliser car il est nécessaire d'entrer toutes les données des blocs Covadis.

Tout d'abord, le projet à produire comprend des informations contenues sur la vue en plan d'une piste mais également des informations contenues sur les coupes des profils en travers de cette piste. Ces indications me permettront ensuite de créer et implanter une polygone en 3D. Avant de commencer la rédaction du programme, j'ai examiné la façon de le créer (*les informations à récupérer, à utiliser et dans quel ordre*).

Pour commencer, l'entreprise m'a donné un exemple projet sur lequel je me suis appuyé (*réflexion, écriture et essais du LISP*) sur le projet de l'aménagement d'une piste à Chamonix. Cependant, ce projet ne correspondait pas aux critères de celui qui m'incombait (*surface géographique non adéquate et pas assez d'informations précises*). De ce fait, j'ai pris l'initiative de me baser sur le projet d'aménagement des pistes Sagnes et Gentianes sur la station de VARS (*cf partie 1.6*) sur lequel j'ai travaillé avec l'équipe du bureau.

Chaque projet est répertorié dans un fichier *.dwg (*format de fichier standard d'enregistrement des graphiques vectoriels.*) et comporte les éléments suivants :

- L'ortho-photographie⁽⁴⁾ de la station
- Un extrait de carte IGN⁽⁵⁾ (*Institut géographique national*) du site ;
- Un fond topographique (*il contient les courbes de niveaux du terrain Naturel destinées à donner sur une carte un aperçu du relief réel*);
- Le contour des pistes de ski étudiées ;
- Les aménagements situés autour du projet (*autres pistes du domaine, chemins, remontées mécaniques, relevés géologique*)
- Les réseaux neige et électricité.

Ce fichier comprend aussi les dessins - réalisés grâce à Covadis - des profils en long et profils en travers des pistes à aménager et les zones de déblais et remblais ; Ils sont représentés sur une vue en plan (plan d'origine réalisé grâce au relevé topographique du géomètre) et une vue en coupe pour chaque profil (*réalisé grâce à Covadis*).

(1) **Covadis** : Applicatif d'AutoCAD en topographie, cartographie, terrassement, infrastructure et projets VRD.

(2) **Blocs** : Terme générique désignant un ou plusieurs objets combinés pour créer un objet unique.

(3) **Polygones** : Objet composé d'un ou de plusieurs segments de droite ou arcs reliés entre eux, et considérés comme un seul objet. Une polygone est considérée comme une seule ligne.

(4) **Ortho-photographie** : Document photographique obtenu par redressement, mise à l'échelle et assemblage des surfaces élémentaires de photographies aériennes.

(5) **Carte IGN** : Représentation à moyenne ou petite échelle des éléments naturels et artificiels situés sur la surface terrestre, ainsi que des formes du terrain.

Sur la vue en plan, les profils en travers sont de simples blocs représentés par une ligne avec un numéro à une extrémité (*chaque bloc contient plusieurs attributs*⁽⁶⁾ : un numéro de profil en long, un numéro de profil en travers, la distance cumulée de l'emplacement du profil en travers, le maintien de l'axe, celui du segment, le type de paramétrage et l'épaisseur de décapage).

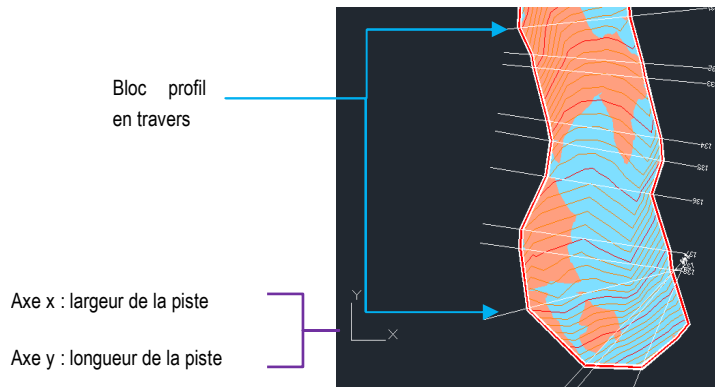


Figure 16 : Vue en plan projet

Disposées à côté de la vue en plan, les coupes des profils en travers⁽⁷⁾ tracés sur cette première laisse apparaissent le terrain naturel, le projet de la piste, son axe (*profil en long*⁽⁸⁾) et le bloc de plan de comparaison⁽⁹⁾ du profil. Ce bloc contient des attributs correspondant à ceux de la vue en plan, comme le numéro de profil en long ; le numéro de profil en travers et également des attributs propres à chaque coupe : l'échelle horizontale et verticale, le plan de comparaison (*altitude*) et le calque⁽¹⁰⁾ d'insertion.

Afin de démarrer mon programme conformément à la commande du client, j'ai tout d'abord travaillé sur papier. J'ai réalisé un plan en trois phases.

La première concerne la vue en plan. J'ai relevé les informations qui m'étaient indispensables et, j'ai opté pour récupérer l'orientation du profil, le numéro du profil en long, celui en travers et le point d'insertion du bloc. Ces informations sont des attributs contenus dans le bloc du profil en travers (*PT*) ; Il faut donc les extraire. Le point d'insertion, quant-à lui, est le point de départ du bloc (point de départ de la ligne), il faut donc enregistrer l'abscisse et l'ordonnée de ce point de départ.

J'utilise les coordonnées du point d'insertion du bloc (*milieu du PT*), l'adresse du point de départ du bloc et la longueur de la ligne du bloc. Avec ces renseignements je calcule les coordonnées du point d'arrivée de la ligne du PT et son orientation en radian, grades ou degrés.

J'utilise un programme permettant d'extraire et d'enregistrer les attributs d'un bloc dans une liste (je peux ainsi sélectionner dans cette liste les attributs qui m'intéressent, en l'occurrence le numéro de profil en long et en travers). Je stocke dans des variables spécifiques toutes les informations recueillies, dans le but de les utiliser lors de la phase d'implantation (phase 3).

⁽⁶⁾ **Attributs** : C'est une entité textuelle définie par l'utilisateur utilisée pour le stockage de données. Un attribut de bloc est un membre imbriqué d'une définition de bloc AutoCAD et peut être configuré pour être visible ou invisible.

⁽⁷⁾ **Profil en travers** : Représentation par une coupe du terrain naturel verticale perpendiculaire de l'axe (*profil en long*) du projet

⁽⁸⁾ **Profil en long** : Représentation d'une coupe verticale du terrain naturel suivant l'axe d'un projet linéaire.

⁽⁹⁾ **Plan de comparaison** : C'est l'axe des abscisses du graphique sur lequel sont reportées les distances horizontales suivant l'axe du projet. (*Plan horizontal de référence*).

⁽¹⁰⁾ **Calque** : Regroupement logique de données comparable à des calques superposables sur un dessin. Vous pouvez visualiser les calques séparément ou en combinaison.

Lors la seconde étapes, l'opération est identique à la première, mais cependant elle concerne les coupes de profil en travers.

J'ai choisi de récupérer dans le bloc de comparaison, le numéro de profil en long, le numéro de profil en travers et l'altitude du plan de comparaison. Les numéros de profil en long et en travers me serviront à vérifier que l'utilisateur a bien récupéré les mêmes blocs (*bloc de profil en travers en relation avec sa coupe de profil en travers*). Ces informations sont des attributs contenus dans le bloc, il faut donc les extraire. Par conséquent, j'ai dupliqué le programme de la phase 1 en changeant simplement le nom des variables.

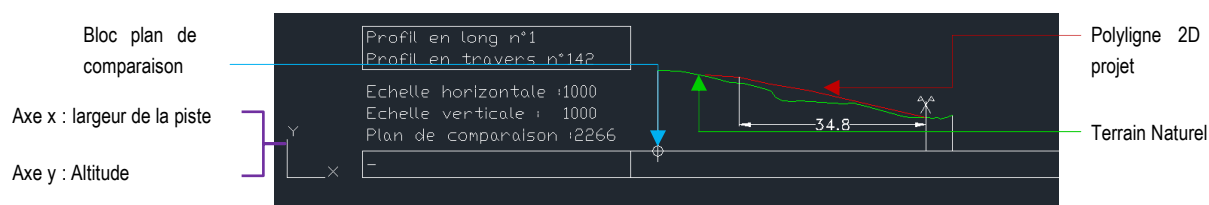


Figure 17 : Vue en coupe du projet (Profil en travers)

Quant à la troisième phase il s'agit de récupérer les informations de la polyligne 2D projet de la coupe du profil et de la transformer en polyligne 3D grâce aux renseignements de la polyligne 2D et ceux des deux premières étapes.

Pour cela il me faut enregistrer dans des variables les coordonnées de chaque sommet de la polyligne 2D du profil étudié afin de pouvoir calculer leurs altitudes et la distance les séparant du plan de comparaison (PDC).

Le plan de comparaison (*voir figure ci-dessus*) représente l'altitude de comparaison et également le début de la ligne du bloc profil en travers sur la vue en plan.

J'aurai pu utiliser un changement de scu⁽¹¹⁾, car le repère n'est pas le même sur la vue en plan et la coupe du profil en travers ; mais j'ai préféré utiliser des calculs simples.

Sur la coupe du profil en travers, l'abscisse indique la largeur de la piste, l'ordonnée l'altitude, alors que sur la vue en plan, l'ordonnée représente la longueur de la piste.

Il me suffit d'additionner, la dénivelée (*entre le plan de comparaison et le premier sommet de la polyligne 2D*) à l'altitude du plan de comparaison (*valeur de l'attribut*) pour trouver l'altitude du sommet.

Grâce à l'orientation du profil en travers et à la distance séparant le PDC du premier sommet, je peux obtenir les coordonnées du premier sommet. Je répète cette formule pour tous les sommets de la polyligne 2D et j'enregistre les coordonnées x, y et z de chaque nouveau sommet dans de nouvelles variables afin d'implanter des points à ces coordonnées.

Une fois les points implantés, je peux tracer une polyligne 3D reliant les points du profil en travers étudié. Lorsque la polyligne est tracée, une boucle permet de dessiner un autre profil ou de terminer le programme.

⁽¹¹⁾ **SCU** : Système de coordonnées défini par l'utilisateur qui spécifie l'orientation des axes X, Y et Z dans un espace 3D. Le SCU détermine la position par défaut des objets dans un dessin.

Je me suis ensuite demandé, dans quel calque, seront intégrés les points et les polygones. J'ai donc décidé de créer un calque projet, dans lequel serait enregistré les points et polygones 3D créés. Je laisse à l'utilisateur la possibilité d'ajouter, s'il le souhaite, un suffixe et un numéro au nom du calque prédéfini et que j'appel « PRO - ». Exemple « PRO – PISTE2 ».

Arrivé à ce point, il m'a semblé important d'inclure une notice d'accompagnement (*message box*) lors de l'emploi du programme, pour permettre aux projeteurs d'utiliser à bon escient le logiciel en sélectionnant les bons objets (*blocs et polygones*) ; si les profils ne correspondent pas un message alerte s'affiche. Par exemple, lors de la sélection du bloc de profil en travers, si l'entité sélectionnée est autre qu'un bloc de profil en travers, le message apparaît expliquant l'erreur commise, ce que doit modifier l'utilisateur et, d'où l'utilisateur doit reprendre le programme.

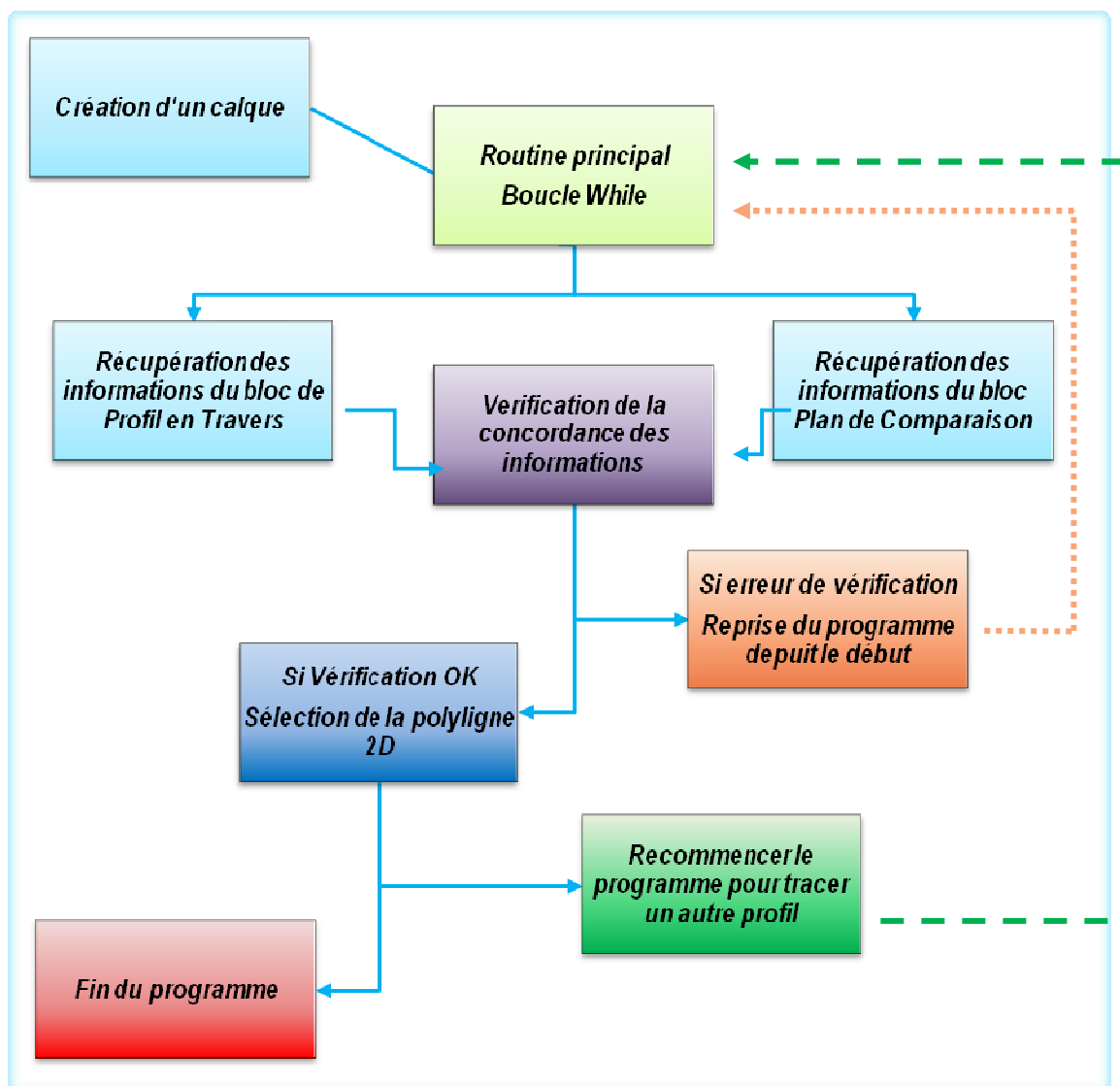
3. Ecriture du LISP

3.1. Premiers essais

Ma première écriture de ce programme contenait la création d'un seul calque dans lequel était enregistré les points et polygones créés, le stock de renseignements sur la vue en plan d'un profil en travers et de sa coupe. Celui-ci était trop basic et son rendement trop faible.

Je l'ai donc amélioré, en y intégrant des vérifications de sélections effectuées et d'une boucle permettant l'enchaînement de profil en travers. Dans les paragraphes suivants, je vous expliquerai la méthode utilisée.

Ci-dessous un processus expliquant l'enchaînement des actions réalisées par l'utilisateur.



ROUTINE PRINCIPAL

La routine principale contient une boucle « while », les appels des sous routines dans lesquelles les calculs sont réalisés ainsi que la vérification de la concordance.

La boucle « while » permet, si on le souhaite, de tracer des profils en boucle, une fois qu'un profil est terminé le programme propose de tracer un autre profil ou de terminer le programme. Cette boucle permet si on tape au clavier « s » (*Suivant*) d'enchaîner les tracés de profils sans recommencer le programme depuis le début et donc de gagner du temps.

Si on tape « T » (*Terminer*) le programme se termine et un message donne le nombre de profils tracés.

ROUTINE n°1 : Création de calques

Afin de ranger correctement les points et les polygones créés avec ce programme, un calque est généré. La fonction « getstring », permet de demander à l'utilisateur de nommer le calque (*exemple : « PISTE »*) et, de donner, le numéro de profil en long étudié (*exemple : « 1 »*) afin de différencier les calques en cas de nombreux profils en long à exécuter.

Les fonctions, commençant par « GETxxx » renseignent une saisie de données (*au clavier ou directement dans autocad*). En ce qui concerne les « xxx », ils représentent le type de données à saisir (*distance, angle, point, nombre, texte ou variables systèmes d'Autocad*). De ce fait, le calque se nommerait « PRO – PISTE1 ». Il est alors créé grâce à la formule « entmake ». Un message donne le nom du calque dans les lignes de commande⁽¹²⁾. Lorsque celui-ci est produit, nous pouvons commencer la récupération d'informations sur le bloc de profil en travers.

ROUTINE n°2 : Récupération d'informations dans le bloc de profil en travers

⇒ [Vue en plan](#)

Je me suis tout d'abord trompé de voie ; par conséquent, j'ai décidé d'exposer le bloc de profil en travers afin de pouvoir récupérer son orientation, son point d'insertion et sa longueur pour l'implantation de ma polyligne 3D.

Cependant, je me suis vite aperçu qu'explorer chaque bloc prenait un temps considérable et le résultat final n'était pas concluant. Je pouvais récupérer les informations contenues dans un bloc en réalisant un simple clique avec la fonction « nentsel ».

En effet celle-ci permet de nommer une entité, pour ensuite, avoir accès aux éléments de l'objet (*base de données d'Autocad*).

Une fois le bloc enregistré, il ne reste plus qu'à trier les indications qui me sont nécessaires ; pour cela j'utilise les opérateurs « car » et « cdr ». (*Opérateur de traitement de liste, « car » permet de prendre le premier élément d'une liste et « cdr » rend une nouvelle liste privée de son premier élément. Il est possible de combiner les deux opérateurs entre eux afin de récupérer n'importe quel élément d'une liste.*)

⁽¹²⁾ **Ligne de commande** : Zone de texte réservée aux entrées clavier, aux invites et aux messages.

Je réalise ensuite des calculs, afin de créer les nouveaux sommets de la polyligne 3D. Les informations contenues dans le bloc me permettent de sauvegarder les coordonnées des points de départ, d'insertion et d'arrivée, la longueur et l'orientation de la ligne du profil en travers. Je m'attèle ensuite sur l'extraction d'attributs de bloc.

Je me sers d'une routine qui enregistre, lorsque l'on clique sur un bloc, tous ces attributs dans une liste. Il ne reste qu'à sélectionner les attributs qui nous intéressent grâce aux opérateurs « car » et « cdr ». Je récupère donc, deux attributs : « PT_NPL » (*numéro de profil en long*) et « PT_NUM » (*numéro de profil en travers*). Ces informations (*nom et valeur*) s'affichent dans la barre de commande grâce aux fonctions « prompt », « princ » ou « print », afin que l'utilisateur puisse voir et vérifier les numéros du profil sélectionné.

ROUTINE n°3 : Récupération d'informations dans le bloc de plan de comparaison

⇒ [Vue en coupe](#)

La routine permettant de récupérer les renseignements contenus dans le bloc PDC est la même que celle pour le bloc profil en travers, seul, quelques variables sont différentes, mais le principe est le même. Le point d'insertion et les attributs du bloc sont les données qui me sont nécessaires.

Comme pour la routine n°2 je sauvegarde le numéro de profil en long et le numéro de profil en travers ; seulement j'enregistre également l'attribut donnant l'altitude du plan de comparaison « PT_P_CMP » qui me permet de calculer les altitudes des sommets de la polyligne 3D.

Grâce à ces attributs, je peux par conséquent vérifier si le bloc PT et le bloc PDC ont les mêmes numéros (*même numéro profil en travers et même numéro de profil en long*).

Dans ce cas le programme s'enchaîne sur la sélection de la polyligne2D, sinon, en cas de différence de numéro de profil (en long ou en travers) entre le bloc PT et le bloc PDC, un message d'erreur dans lequel les erreurs de numéro s'affichent.

ROUTINE n°4 : Sélection de la polyligne 2D et implantation des points et de la polyligne 3D

⇒ [Vue en coupe](#)

Pour stocker tous les sommets de la polyligne projet 2D représentée sur la coupe du profil en travers, j'utilise la fonction « entsel » afin de sélectionner la polyligne, puis la fonction « mapcar » pour créer une liste dans laquelle les coordonnées x et y des sommets sont enregistrés dans l'ordre croissant des sommets. Ensuite, je crée à partir de ces sommets les nouveaux sommets 3D qui viendront s'implanter sur la vue en plan.

Je récupère l'abscisse et l'ordonnée du premier sommet ; je calcule la dénivelée (y) et la distance horizontale (x) entre le point d'insertion du plan de comparaison et ce sommet.

Grâce, à cette distance horizontale, à l'orientation du bloc et aux coordonnées x et y du point de départ du bloc profil en travers, je calcule les coordonnées x et y du premier point (*sommet*) de la polyligne 3D, et également l'altitude de ce point (*altitude du plan de comparaison ajouter à la dénivelée calculée*); j'ajoute l'altitude aux coordonnées du sommet de façon à me retrouver avec des coordonnées x, y, z.

Je réalise cette opération vingt fois, de façon à pouvoir implanter une polyligne de vingt sommets au maximum.

A partir de ce moment il n'y a plus de calcul, seule une fonction « if » est à créer afin d'implanter les points calculés précédemment. Elle fonctionne de la façon suivante : si il y a deux sommets sur la polyligne ; alors deux points sont implantés aux coordonnées calculées (sommets) ; une polyligne 3D est tracée du premier point au second ; et les deux points et la polyligne sont placés dans le calque prévu à cet effet, sinon la fonction « if » se termine et passe à la fonction « if » suivante avec trois sommets.

Si la polyligne 2D contient trois sommets, le programme implante les trois points et trace la polyligne entre ces points ; sinon il passe à la fonction d'après avec quatre sommets et ainsi de suite jusqu'au vingtième.

Si la polyligne 2D de projet contient plus de vingt sommets, alors le programme ne trace que les vingt premiers sommets (*il arrive rarement d'avoir plus de vingt sommets sur une polyligne de projet*).

Un message apparaît à chaque fin de profil réalisé, pour donner le nombre de points créés et donc, le nombre de sommets contenus dans la polyligne créée.

Lorsque l'on choisi de terminer le programme, un message s'affiche également révélant le nombre de profils tracés (*nombre de boucle « while » effectué*).

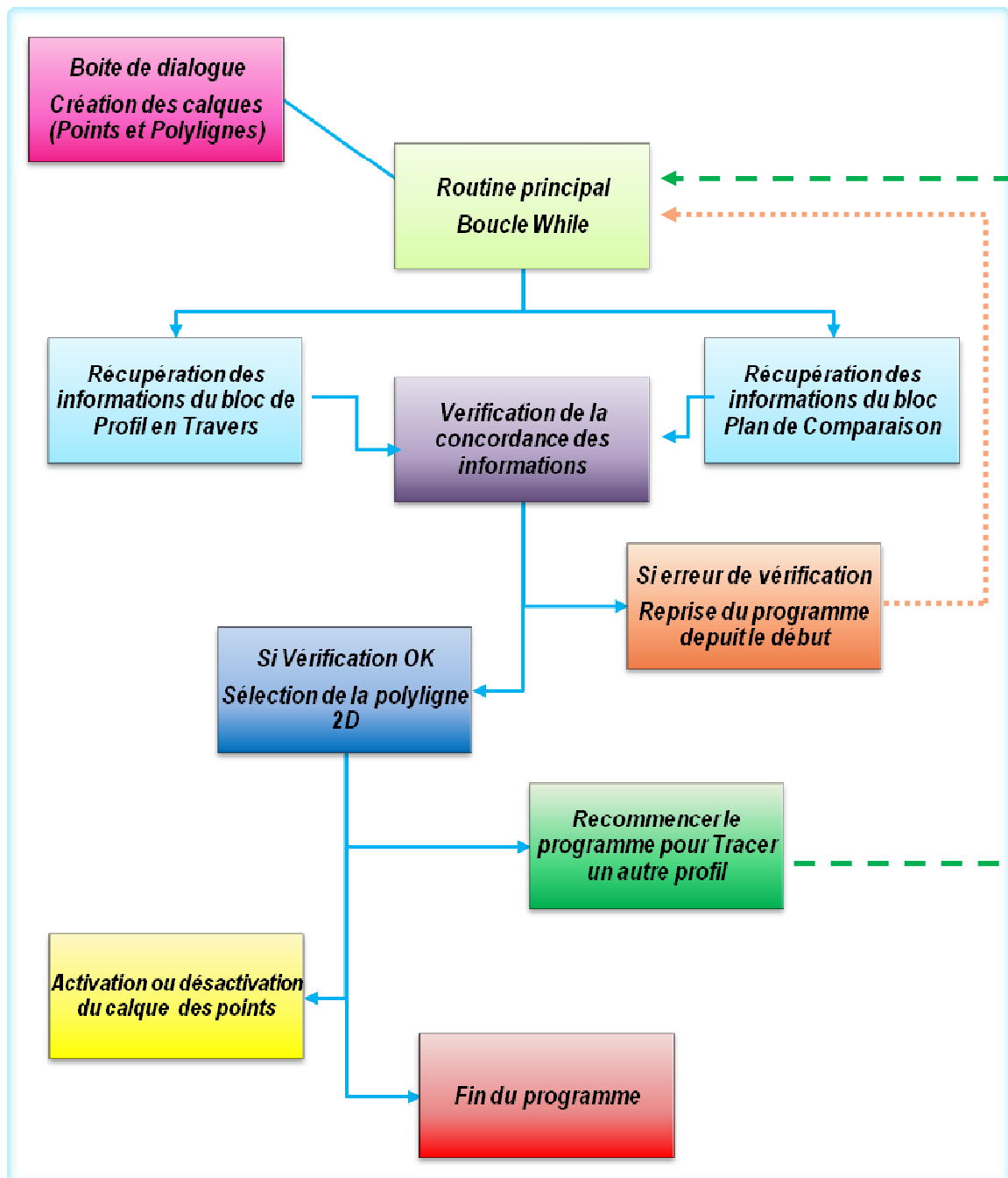
A ce stade le programme tournait bien, mais celui-ci était trop long et manquait de vérification en cas d'erreur de sélection. Il y avait trop de paramètres à rentrer au clavier. En conséquence, mon premier programme devait être amélioré, raccourci et simplifié.

L'approche et l'utilisation de programme allaient être plus performantes.

3.2. Évolution du programme

J'ai donc repris le code programmé précédemment, bien que celui-ci était tout à fait correct. Une nouvelle problématique se dressait : par quels moyens puis-je simplifier et rajouter quelques commandes utiles au programme afin de gagner du temps.

Ci-dessous le processus du programme final :



Après réflexion, j'opte pour l'intégration d'une boîte de dialogue en début de programme (message box), de façon à créer ou à choisir les deux calques nécessaires. Cette aide imagée et légendée permet également d'expliquer les actions à mener pour exécuter le programme. J'ai, de plus, ajouté en fin de programme, si l'utilisateur le souhaite, la possibilité de désactiver le calque contenant les points du projet, de façon à ne laisser apparaître que les polygones 3D pour, plus de lisibilité.

Ci-dessous la boîte de dialogue à l'état final :

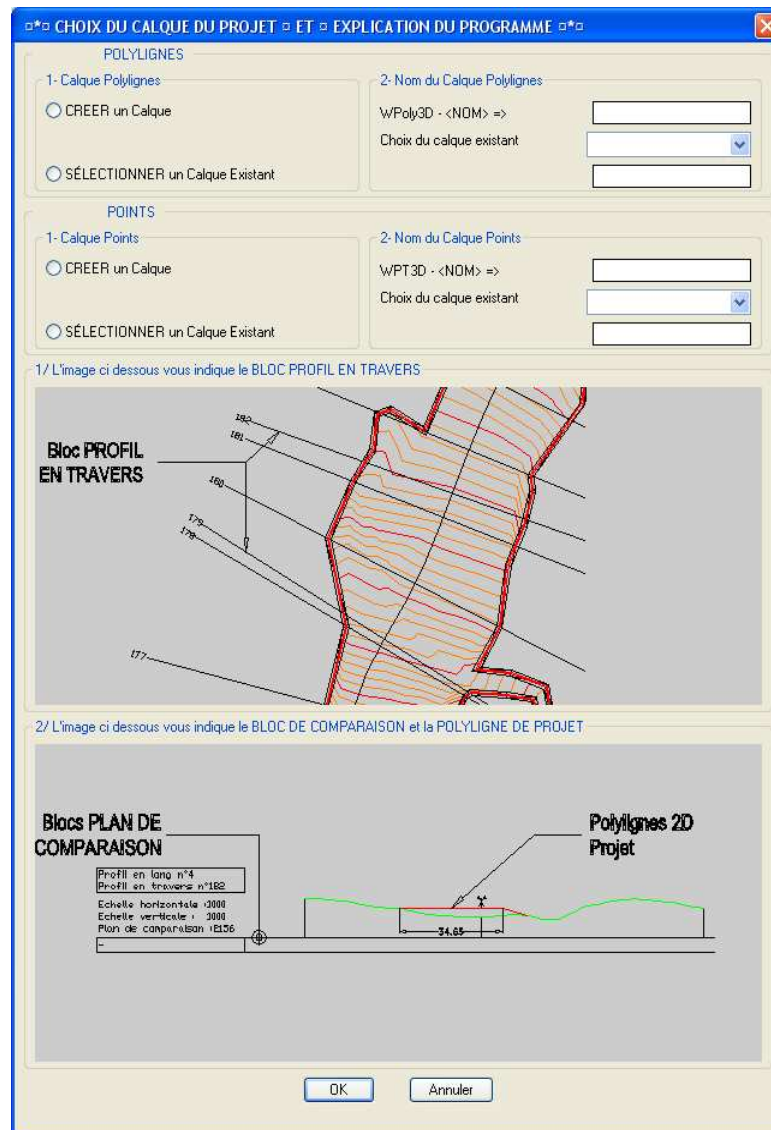


Figure 18 : La boîte de dialogue

Routine principale T :

```
(defun c:t ()
  (vl-load-com)
  (setq "OSMODE" 1)
  (setq "CMDECHO" 0)
  (RemiseAZero)

  (calque)

  (boucletrace)

  (setq geler (getstring
    "\nVoulez vous désactiver le calque des points ? <Oui/Non>"
  ))
  (if (or (= geler "0") (= geler "o"))
    (vlax-for lay (vla-get-Layers
      (vla-get-ActiveDocument (vlax-get-acad-object)))
      (if (wcmatch (strcase (vla-get-Name lay)) (strcase namep))
        (vla-put-LayerOn lay :vlax=false)
      )
    )
  )
  (princ)
)
```

Figure 19 : Routine principale

La routine⁽¹³⁾ « t », est le programme principal, elle permet de mettre les variables, « OSMODE » qui définit les modes d'accrochage aux objets, et « CMDECHO » qui définit les invites⁽¹⁴⁾ et les entrées reprises lors de la fonction de commande AutoLISP.

Le programme permet aussi d'appeler les routines (*RemiseAZero*), (*calque*), (*boucletrace*). Il donne par ailleurs la possibilité de désactiver le calque des points du projet à la fin du programme.

Routine (*RemiseAZero*) :

Cette routine donne la possibilité de remettre à zéro les variables devant être vide à chaque lancement pour le bon déroulement du programme et pour obtenir les bons résultats. (*J'aurai pu également écrire ces variables à la première ligne du programme, dans les parenthèses situées après le nom de la routine, et précédés d'un antislash (!) ; j'ai préféré utilisé un programme spécial pour plus de lisibilité.*)

Routine (*calque*) :

Afin de ranger correctement les points et les polygones créés avec ce programme, deux calques sont à concevoir avant le démarrage du programme ; le premier contient les points et le second les polygones.

Pour cette routine, j'emploie une boîte de dialogue qui apparaît devant le dessin afin d'y entrer des valeurs pour optimiser du temps. Elle est liée à un programme permettant de sélectionner des calques et d'expliquer le fonctionnement du programme.

⁽¹³⁾ **Routine** : Partie du code d'un programme informatique destiné à être utilisé plus d'une fois.

⁽¹⁴⁾ **Invites** : Message sur la ligne de commande ou dans une info-bulle invitant l'utilisateur à indiquer des informations ou à exécuter une action, telle que la spécification d'un point.

Ci-dessous, le fichier « DCL », où l'on crée l'environnement de la boîte de dialogue : le visuel.

```

calque1:dialog {label=" *** CHOIX DU CALQUE DU PROJET * ET * EXPLICATION DU PROGRAMME *** ";
:boxed_row {label="\t                POLYLIGNES  ";
:boxed_radio_column {label="1- Calque Poly lignes";
:radio_button {label="CREER un Calque";key="nouveauC";value="0";}
:radio_button {label="SÉLECTIONNER un Calque Existant";key="courantC";value="0";}}
:boxed_radio_column {label="2- Nom du Calque Poly lignes";
:radio_column {key = "Selectionducalque";
:edit_box {label="WPoly3D - <NOM> =>";key="newc";edit_width=20;fixed_width=false;}
:popup_list {key="liste";label = "Choix du calque existant";edit_width = 20;}
:edit_box {key="NOMDUCALQUE";edit_width = 20;}
}}
}}
:boxed_row {label="\t                POINTS  ";
:boxed_radio_column {label="1- Calque Points";
:radio_button {label="CREER un Calque";key="nouveauCP";value="0";}
:radio_button {label="SÉLECTIONNER un Calque Existant";key="courantCP";value="0";}}
:boxed_row {label="2- Nom du Calque Points";
:radio_column {key = "Selectionducalquep";
:edit_box {label="WPT3D - <NOM> =>";key="newcp";edit_width=20;fixed_width=false;}
:popup_list {key="listep";label = "Choix du calque existant";edit_width = 20;}
:edit_box { key="NOMDUCALQUEP";edit_width= 20;}
}}
}
:row {
:boxed_column {label="1/ L'image ci dessous vous indique le BLOC PROFIL EN TRAVERS";
:image {width=20;aspect_ratio=1;height=20;key=im1;}
}
}
:row {
:boxed_column {label="2/ L'image ci dessous vous indique le BLOC DE COMPARAISON et la POLYLIGNE DE PROJET";
:image {width=20;aspect_ratio=1;height=20;key=im2;}
}
}
:row {ok_cancel;}
:row {errtile;}
}

```

Figure 20 : Organisation de la boîte de dialogue

Les accolades ci-dessus représentent :

- 1) La création ou sélection du calque qui contiendra les polygones 3D du projet.
- 2) La création ou sélection du calque qui contiendra les points du projet.
- 3) L'affichage d'une image légendé indiquant ce qu'est un bloc profil en travers.
- 4) L'affichage d'une image légendé indiquant ce qu'est un bloc plan de comparaison et la polygones projet.

Une boîte de dialogue composée de quatre lignes a été mise au point. Les deux premières lignes sont décomposées en deux colonnes, consacrées à la création ou sélection de calques.

Dans la première colonne de la première ligne, consacrée au calque des polygones3D, deux « radio_button » (il s'agit d'une catégorie de bouton grâce auquel, lorsque l'on a plusieurs possibilités on ne peut en choisir qu'une, dans notre cas : soit on crée un nouveau calque soit on sélectionne un calque existant) sont insérés.

Dans la deuxième colonne de la première ligne, se trouve une case permettant de taper le suffixe du nom du calque qui contiendra les polygones du projet. Le préfixe de ce calque est « WPoly3D - » et est invariable. Dans cette colonne se trouve également une « popup_list » (il s'agit d'une liste déroulante, dans notre cas la liste des calques existants dans le dessin dans lequel nous travaillons. Attention, si le calque est gelé⁽¹⁵⁾

dans l'espace objet, la sélection du calque dans la liste est impossible et entraîne une erreur ainsi que la fin du programme).

A la deuxième ligne, il s'agit du même programme que précédemment pour le calque des points du projet. Son préfixe en cas de création est « WPT3D - » ; en cas de sélection une « popup_list » est aussi présente.

Lorsque l'on clique sur un radio_button, le lisp va recevoir la clé (key) de la boîte de dialogue, qui va lui indiquer de lancer le code.

Les deux dernières lignes (accolades verte et violette) du DCL sont consacrées à l'insertion d'images (clichés⁽¹⁶⁾ Autocad). Celles-ci montrent les objets à sélectionner pour réaliser la suite du programme. Sur le premier cliché se trouve une vue en plan (sur laquelle une flèche associée à sa légende pointe le bloc profil en travers à sélectionner après la fermeture de la boîte de dialogue). Quant-au deuxième cliché, représentant la coupe du profil en travers, deux flèches associées à leur légende pointent le bloc plan de comparaison et une polyligne 2D projet. (Ces derniers sont à choisir après avoir sélectionné le bloc profil travers).

Dans le lisp

```
(defun calque ()
  (setq dcl_id (load_dialog "calque.dcl"))
  ;;charge la boîte de dialogue
  (if (not (new_dialog "calque1" dcl_id)
      ;;si il ne la trouve pas il sort "exit"
      )
      (exit)
  )
)
```

Figure 21 : Chargement boîte de dialogue

Les trois premières lignes permettent de charger la boîte de dialogue. Si le programme ne la trouve pas il se termine, sinon la boîte de dialogue apparaît à l'écran.

Ensuite les variables nécessaires à la création ou à la sélection du calque sont remis à zéro. La suite de la routine fonctionne avec la boîte de dialogue.

Lorsque l'on utilise des « radio_button », on a simplement besoin de signifier quelle action doit être lancée lorsque la clé est envoyée au lisp.

```
(action_tile "nouveauC" "(nouveau calque)")
(action_tile "courantC" "(courant calque)")
(action_tile "nouveauCP" "(nouveau calque)")
(action_tile "courantCP" "(courant calque)")

(action_tile "accept" "(done_dialog 1)")
(action_tile "cancel" "(done_dialog 0)")
```

Figure 22 : Actions lancées

L'« action_tile » définit l'action du lisp lorsque la clé est envoyée par le DCL, quand l'utilisateur actionne le radio_button ou la popup_list ou le bouton « Ok ».

⁽¹⁵⁾ **Geler** : Paramètre supprimant l'affichage des objets sur les calques sélectionnés. Les objets se trouvant sur les calques gelés ne sont ni affichés, ni régénérés, ni tracés.

⁽¹⁶⁾ **Clichés** : Vues d'espace objet dans un dessin, enregistré par Autocad. Ces images peuvent apparaître dans les boîtes de dialogue.

Si le bouton “créer un calque” est enclenché alors le programme (*nouveaucalque*) est lancé pour y affecter les polygones ; la boîte dialogue déverrouille une case « edit_box » permettant de taper du texte (*il faut entrer le suffixe du nom du calque à créer*) et verrouille également la « popup_list » permettant la sélection d’un calque existant.

Au contraire, si le bouton “sélectionner un calque existant” est actionné alors le programme (*courantcalque*) est lancé pour y affecter les polygones ; la boîte dialogue déverrouille alors la case « popup_list » pour sélectionner le calque existant dans la liste déroulante et ferme la case « edit_box » permettant de noter le suffixe du calque.

```
(START_IMAGE "im1")
;;Fonction qui permet de charger une image
(FILL_IMAGE 0 0 (DIMX_TILE "im1") (DIMY_TILE "im1") 254)
;;image affichée à partir du point 0,0. Le "1" à la fin correspond à la couleur de fond (ici 1=rouge)
(SLIDE_IMAGE
  0
  0
  (DIMX_TILE "im1")
  (DIMY_TILE "im1")
  "im1.sld"
)
(END_IMAGE)
;;fin de l'image
```

Figure 23 : Chargement de l’image 1

Le programme ci-dessus charge le cliché de la première image et définit son emplacement dans la colonne dans laquelle elle sera insérée dans boîte de dialogue

Le même programme est utilisé pour l’affectation des points du projet à un calque. Seul le nom des variables est différent.

Afin de valider les choix effectués, il est nécessaire d’appuyer sur le bouton « Ok » ou de taper sur « entrée ».

L’utilisation des boutons « Annuler » ou « echap » entraînera un non enregistrement des choix suscités ci-dessus. (*Par défaut, la sélection ou la création de deux calques se fera automatiquement*).

Sous routine (*nouveaucalque*) :

```
(defun nouveaucalque ()
  (setq choice "No1")
  (node_tile "newc" 0)
  (node_tile "liste" 1)
  (node_tile "NONDUCALQUE" 1)
  (action_tile "newc" "(setq newc$ $value)")
)
```

Figure 24 : Routine (*nouveaucalque*)

J’affecte la valeur de la clé sélectionnée (*newc\$*) à la variable « newc » préenregistré dans la case « edit_box ». La fonction « mode_tile » permet de verrouiller (position 1) ou déverrouiller (position 0) les variables.

Sous routine (*courantcalque*) :

```
(defun courantcalque ()
  (setq choice "No2")
  (mode_tile "liste" 0)
  (mode_tile "NOMDUCALQUE" 0)
  (mode_tile "newc" 1)
  (setq lb (list))
  (setq lb (cons (cdr (assoc 2 (tblnext "LAYER" T))) lb))
  (while
    (setq X (cdr (assoc 2 (tblnext "LAYER"))))
    (setq lb (cons X lb))
  )
  (setq listecalqueordre (acad_strlsort lb))
  (start_list "liste")
  (mapcar 'add_list listecalqueordre)
  (end_list)
  (action_tile
    "liste"
    "(setq calquechoix (nth (atoi $value) (acad_strlsort lb))) (set_tile \"NOMDUCALQUE\" calquechoix)"
  )
)
```

Figure 25 : Routine (*courantcalque*)

Dans ce cas, je rentre la valeur de la clé prédéfinie (*liste*) à la variable « *NOMDUCALQUE* » qui est le calque sélectionné dans la « *popup_list* ». J'enregistre dans la variable « *lb* » la liste des calques du dessin pour les utiliser dans ma liste déroulante, pour cela j'affecte à la variable « *liste* », la liste des calques, alphabétiquement, qui apparaîtra dans ma boîte de dialogue.

La dernière ligne « *action_tile* » me permet de récupérer le nom du calque sélectionné dans la liste de la boîte.

Une fois la boîte de dialogue renseignée et fermée, le programme de création ou de sélection (*choixcalque*) du calque des polygones 3D est lancé. Si l'utilisateur ne fait aucun des deux choix proposés (création ou sélection), le programme crée automatiquement un calque nommé « *WPT3D - 1* » contenant les points et un autre nommé « *WPoly3D - 1* » stockant les polygones. Si ces calques existent déjà, alors la programmation les sélectionne automatiquement.

Sous routine (choixcalque) :

```
(defun choixcalque ()
  (if (and (= newc$ nil) (/= calquechoix nil))
      (progn
        (setq name (strcat calquechoix))
        (setvar "clayer" name)
        (princ (strcat "\nLe calque "
                      calquechoix
                      " a été sélectionné pour contenir les polygones 3D du projet.\n")))
      (progn
        (princ)
      )
    )
  (if (and (/= newc$ nil) (= calquechoix nil))
      (progn
        (if (tblsearch "layer"
                      (setq name (strcat "WPOLY3D - " newc$)))
            )
        (princ
         (strcat "\nCe calque =>>>" name "<<<= est déjà existant.")
        )
        (progn
          (entmake (list (cons 0 "layer")
                        (cons 100 "AcDbSymbolTableRecord")
                        (cons 100 "AcDbLayerTableRecord")
                        (cons 2 name)
                        (cons 6 "CONTINUOUS")
                        (cons 62 110)
                        (cons 70 64))))
          (setvar "clayer" name)
          (princ (strcat "\nLe calque "
                        name
                        " a été créé pour contenir les polygones 3D du projet.\n")))
        (progn
          (princ)
        )
      )
    )
  (princ)
)
```

Figure 26 : Routine (choixcalque)

Concernant cette partie du programme, la fonction « if » permet d'effectuer un seul des deux programmes proposés.

Soit le calque est à créer, soit il est existant. Suivant le choix, la fonction « if » décline un programme différent. L'exemple ci-dessus, montre l'affectation de la variable « name » au nom du calque dans lequel seront enregistrées les polygones 3D créées.

La première ligne du programme de la fonction « if » réalise un test logique, deux possibilités s'offrent :

- si le résultat du test est vrai : alors la variable « name » enregistre le nom du calque sélectionné dans la liste dans la boîte de dialogue
- sinon la fonction « if » se ferme et propose l'option suivante : le programme cherche si un calque de même nom n'est pas déjà présent dans le dessin ; s'il existe un message s'affiche, sinon le calque est créé par défaut (*nom du calque, couleur, type de traits, épaisseur de traits...*).

Une fois de plus, un programme quasi identique est lancé pour le calque des points.

Le troisième, appelé sur deux premiers, effectue l'affectation d'un calque aux objets créés lorsque l'on ne sélectionne rien dans la boîte de dialogue.

Sous routine (*OrientationEtCoordonneesBloc*) :

⇒ [Vue en plan](#)

```
(defun OrientationEtCoordonneesBloc ()
  (setq BLOC
    (nentsel
      "\n>>> VUE EN PLAN \nSelectionner le bloc du Profil en Travers sur la VUE EN PLAN : "
    )
  )
  ;; Selection du bloc et enregistrement des données du bloc dans "BLOC"
  (setq a (car (cdr (cdr BLOC))))
  ;; Récupération de la liste du BLOC qui nous interesse
```

Figure 28 : Sélection PT et enregistrement de ses informations

La routine (*OrientationEtCoordonneesBloc*) récupère les informations contenues dans un bloc PT sur la vue en plan. La fonction « nentsel » nomme une entité unique (*ici le bloc profil en travers*), à partir de ce nom nous avons accès aux caractéristiques de cette entité grâce aux opérateurs « car » et « cdr ». Ces opérateurs permettent d'enregistrer dans de nouvelles variables, les coordonnées du point d'insertion du bloc (*milieu de la ligne du bloc*), la base, la dénivelée et la longueur de la ligne du bloc.

Si l'entité sélectionnée n'est pas un bloc PT alors un message d'erreur apparaît, expliquant l'erreur commise. Grâce à quelques calculs simples avec les informations de ce bloc, les coordonnées du point de départ, du point d'arrivée et l'orientation de la ligne du bloc sont obtenus. La longueur de la moitié de la ligne du PT et les coordonnées du point de départ du PT sont également calculées.

La sous routine (*GetAttribsc*) (*récupérée sur un forum*), est utilisé pour extraire au bloc les attributs "PT_NUM" et "PT_NPL", représentant respectivement le numéro de profil en travers et le numéro de profil en long.

Une nouvelle vérification, s'effectue dès lors sous forme de fonction « if », permettant de contrôler si le numéro de PL attribué à la variable « NumeroPLC » est vide. Si oui, une alerte se déclenche ; sinon le programme se poursuit.

Sous routine (*extractionPDC*) :

⇒ [Vue en coupe](#)

Elle ressemble à la routine précédente, et permet de choisir (*par clique*) une entité unique (*bloc PDC*) afin de récupérer les informations incluses.

Si l'entité choisie n'est pas un bloc PDC, un message identique à la routine (*OrientationEtCoordonneesBloc*) apparaît, sinon le programme enregistre dans la variable « xPDC » l'abscisse du point d'insertion de ce bloc PDC, et dans « yPDC » son ordonnée. J'appelle ensuite la routine extrayant les attributs du bloc.

Je récupère ainsi le numéro de profil en travers, le numéro de profil en long et l'altitude du point du plan de comparaison que je stocke respectivement dans les variables « numeroPTP », « numeroPLP » et « PDC ».

```

(if (= PDC nil)
  (progn
    (setq messag1
      (strcat
        "\n\t\t * * * * * "
        "\n\t\t LA SELECTION QUE VOUS AVEZ FAITE N'EST PAS BONNE !"
        "\n\t\t Vérifiez que vous avez sélectionné le bon bloc (BLOC DE COMPARAISON sur VUE EN COUPE). "
        "\n\t\t => Veuillez recommencer votre selection de Profil sur la VUE EN PLAN \n => puis sur la VUE EN COUPE "
      )
    )
    (alert messag1)
    (boucletrace)
  )
)

```

Figure 29 : Message d'erreur

Le programme ci-dessus permet de vérifier que l'altitude du plan de comparaison ne soit pas vide (*nil*), si il est vide le message "messag1" apparaît expliquant comme les messages précédents les erreurs commises, et recommence le programme depuis la routine (*boucletrace*) ; sinon le programme lance la routine (*implantation*) permettant de sélectionner la polyligne 2D de la coupe du profil en travers et d'implanter et de tracer les points et la polyligne.

Sous routine (*implantation*) :

⇒ [Vue en coupe](#)

```

(defun implantation ()
  (setq pPDC (list xPDC yPDC PDC))
  (command "luprec" 2)
  (command "lunits" 2)
  (princ
    "\n>>> COUPE \nSelectionner la polyligne du PROFIL EN TRAVERS qui vous interesse :")
  )
  (setq listesommet
    (mapcar 'cdr
      (vl-remove-if-not
        '(lambda (x) (= (car x) 10))
        (entget (car (entsel))))
      )
    )
  )
  (setq A1 (car listesommet))
  (setq A2 (cadr listesommet))
  (setq A3 (caddr listesommet))
)

```

Figure 30 : Sélection polyligne 2D et enregistrement de ses sommets

J'affecte tout d'abord à la variable « pPDC » les coordonnées X=xPDC, Y=yPDC et Z=alt du point d'insertion du bloc plan de comparaison.

Je demande à l'utilisateur, par le biais de la fonction « entsel », de sélectionner la polyligne 2D du projet dessiné sur la coupe du profil en travers puis la formule « mapcar » enregistre dans une liste que j'appelle "listesommets" tous les sommets de cette polyligne. Le premier sommet est enregistré dans la première variable de la liste, le deuxième sommet dans la deuxième variable, ainsi de suite jusqu'au vingtième sommet car mon programme permet de sélectionner seulement les vingt premiers sommets de la polyligne. Lorsque les sommets sont enregistrés dans leur valeur respective, le programme appelle la routine (*choixsommets*) permettant de créer le bon nombre de sommets.

Je calcule ensuite les coordonnées x, y et z de chaque sommet de la polyligne 3D grâce aux résultats des sélections effectuées lors des routines précédentes, que je stocke dans une variable que je nomme « POINT(+Numéro_du_sommet) ». Les calculs des sommets sont toujours les mêmes, seul le nom des variables change.

Ci-dessous le calcul du point n°1 de la polygline :

```
(defun POINT1 ()
  (setq x1 (car A1))
  (setq y1 (cadr A1))
  (setq dPDCx1 (- x1 (car pPDC)))
  (setq PTdebut (polar PTbloc2 ang0 dPDCx1))
  (setq Z1 (+ (- y1 (cadr pPDC)) PDC))
  (setq PT1 (list (car PTdebut) (cadr PTdebut) Z1))
)
```

Figure 31 : Routine (POINT1) création des coordonnées du point1

La première ligne affecte « x1 » (abscisse du sommet n°1 de la polygline 2D) au premier élément de la liste « A1 ».

La deuxième ligne affecte « y1 » (ordonnée du sommet n°1 de la polygline 2D) au deuxième élément de la liste « A1 ».

La troisième ligne calcule la distance horizontale séparant le point d'insertion du bloc PDC au premier sommet de la polygline 2D et stocke cette valeur dans une variable nommée « dPDCx1 ».

La ligne suivante enregistre dans « PTdebut » les coordonnées X et Y du point 1. Ces coordonnées sont calculées grâce au point de départ de la ligne du bloc PT (ce point est le même que le point d'insertion du bloc de PDC), à l'orientation du PT et à la distance « dPDCx1 ».

La cinquième ligne additionne à l'altitude du plan de comparaison, la dénivelée séparant le point d'insertion du bloc PDC au premier sommet.

La dernière ligne affecte à la variable « PT1 » les coordonnées 3D du sommet n°1 de la polygline 2D de la coupe du profil en travers, sur la vue en plan.

Sous routine (choixsomets) :

```
(defun choixsomets ()
  (if (and
      (/= A1 nil)
      (/= A2 nil)
      (= A3 nil)
    )
    (progn
      (POINT1)
      (POINT2)
      ;; trace la polygline entre les points
      (setvar "CLAYER" name)
      (command "poly3d" PT1 PT2 "")

      (setq nbresom2
        "\nIl y a 2 sommets sur la polygline sélectionné ")
    )
    (alert nbresom2)
    ;; trace les points
    (setvar "CLAYER" namep)
    (command "point" PT1)
    (command "point" PT2)
    (princ)
  )
  (progn
    (princ)
  )
)
```

Figure 32 : Routine (choixsomets)

Cette routine contient vingt fonctions « if », une seule fonction est vraie par polygline car chaque fonction dépend du nombre de sommet que contient cette polygline. Toutes les autres fonctions sont fausses et n'effectuent donc aucuns calculs.

Par exemple, si la polygline sélectionnée contient deux sommets, la fonction qui va être effectuée est celle qui vérifie que la valeur de la troisième variable de la liste "listesomets" (liste contenant les coordonnées des sommets de la polygline 2D) est vide, ou autrement dit que le troisième sommet n'existe pas.

La fonction appelle ensuite les routines « POINT1 » et « POINT2 » afin de calculer les coordonnées 3D des deux extrémités de la polyligne, puis trace la polyligne depuis le premier point jusqu'au second, tout en plaçant cette polyligne dans le calque choisi en début de programme.

Pour finir, les points sont créés à leurs coordonnées respectives et sont placés dans le calque prévu à cet effet en début de programme.

Un message donne le nombre de sommets tracés et que contient la polyligne 3D.

A partir de ce moment, le profil en travers que l'utilisateur souhaitait tracer est modélisé. Le programme termine donc la boucle « while » de la routine (*boucletrace*). C'est à ce moment là, que l'utilisateur a le choix de tracer un nouveau profil en travers ou de terminer le programme.

4. Problèmes rencontrés

4.1. Récupération d'attributs dans la routine Routine OrientationEtCoordonnéesBloc et ExtractionPDC

Après avoir réalisé l'enregistrement des informations propres aux blocs (*point d'insertion, orientation...*), il me fallait vérifier la concordance des deux blocs sélectionnés.

Le plus simple, pour le vérifier, était de sauvegarder les numéros de profils en long et numéros de profils en travers des deux blocs et de les comparer.

J'ai donc eu rapidement besoin d'un programme permettant l'extraction de ces attributs.

Je ne parvenais pas à les enregistrer dans une liste, car je m'initiais à la programmation. Je me suis donc rendu sur internet, sur des forums, afin de trouver un programme qui me permettrait d'extraire ces entités.

Je me suis donc fortement inspiré de programmes existants.

```
(defun GetAttribs (ent / elst lst)
  (setq ent (entnext ent))
  (while (= "ATTRIB" (cdr (assoc 0 (setq elst (entget ent)))))
    (setq lst (cons (cons (cdr (assoc 2 elst)) (cdr (assoc 1 elst)))
                  lst)
          ent (entnext ent))
  )
  (reverse lst)
)
```

Figure 33 : Programme "GetAttribs"

Plus tard au cours de l'année, on m'a confié la réalisation d'un programme permettant l'export d'attribut sous Excel, j'ai donc développé un programme permettant l'extraction d'attribut ce qui m'a facilité la compréhension de celui-ci.

4.2. Orientation du PT dans la routine Routine OrientationEtCoordonnéesBloc

Lorsque j'ai testé mon tout premier programme, il marchait très bien, j'étais fier de mon programme. Il ne fonctionnait pas si bien comparé à celui que je présente aujourd'hui.

Effectivement, lorsque je modélisais des profils en travers retournés, lorsque les pistes tournent à 180° (*numéro de profil en travers placé à droite et écrit à l'envers*), mes points et polygones n'étaient pas créés sur le PT mais dans son prolongement dans vide.

Les points étaient placés à la bonne distance mais il y avait un problème d'angle.

J'ai donc modifié mon programme grâce à l'inversion d'un simple signe.

4.3. Stockage des sommets de la polyligne 2D dans la routine importation

Afin d'enregistrer tous les sommets de la polyligne 2D de la vue en coupe du profil en travers, il me fallait une fonction permettant la récupération et l'enregistrement de tous ces sommets sous forme de boucle ; la fonction enregistre dans la variable "listesommets", les coordonnées du premier sommet puis passe au sommet suivant, dès qu'il n'y a plus de sommet, la fonction s'arrête. La fonction « mapcar » réalise cette opération.

Ne sachant pas l'utiliser, je suis retourné sur des forums pour trouver une fonction similaire de laquelle je pourrais m'inspirer. J'ai trouvé la fonction que je souhaiter, et je l'ai utilisé.

4.4. Sélection calque gelé dans la boîte de dialogue

A force d'utiliser le programme je me suis rendu compte qu'il renvoyait un message d'erreur lorsque l'on sélectionne au début du programme, dans la boîte de dialogue, des calques gelés dans l'espace objet⁽¹⁸⁾. Ce problème est lié, je suppose, au rendu courant⁽¹⁹⁾ du calque. Pour y remédier, j'ai rajouté à la boîte de dialogue, une ligne permettant de prévenir l'utilisateur de sélectionner un calque non gelé.

4.5. Rendu visuel de la boîte de dialogue

Ma première boîte de dialogue était trop étendue, elle était coupée en bas sur les écrans de PC plus petit que celui que j'utilise.

Les images (clichés) étaient mal dimensionnées et illisibles, de plus, la boîte était mal organisée. J'ai donc décidé de la modifier pour qu'elle soit plus lisible, plus facile à utiliser et moins imposante.

⁽¹⁷⁾ **Espace objet** : Un des deux espaces principaux où les objets résident. En règle générale, le modèle géométrique est défini dans un système de coordonnées en 3D appelé espace objet. Une présentation finale des vues et des annotations spécifiques à ce modèle est placée dans l'espace papier.

⁽¹⁸⁾ **Rendre courant** : Définit le calque sélectionné en tant que calque courant. Les objets que vous créez sont dessinés sur le calque courant.

5. Conclusion

Ce projet m'a permis d'explorer la programmation comme jamais je ne l'avais vue auparavant. L'objectif était de diminuer le temps de production d'un dessinateur et, je pense qu'il répond aisément à cette demande.

Ce programme était très intéressant car en plus de son utilité au sein du bureau, il m'a aidé à comprendre toute l'élaboration de pistes de ski et m'apporte une certaine fierté. En effet j'ai eu de nombreux problèmes : bugs, erreurs d'écriture, modifications à apporter dûs au changement d'ordinateur. J'ai donc posé les problèmes à plat pour réfléchir à la façon de les résoudre et y apporter des actions correctives.

De plus, la création de cet applicatif m'a également permis de me perfectionner en programmation LISP, ce qui m'a été très utile lors de la réalisation d'autres programmes pour le bureau, et qui je pense, me sera utile dans mon futur métier de dessinateur.

Cette année en alternance fut extrêmement bénéfique et enrichissante, autant du point de vue professionnel qu'affectif, et, m'aura permis de créer des liens avec des personnes professionnellement intéressantes et érudites. Tout au long de celle-ci, j'ai su conforter et consolider mes connaissances et en ai obtenues de nouvelles. J'ai également acquis en maturité professionnelle et enrichi ma professionnalisation, grâce aux directives que l'on m'a transmises, à l'investissement et au dynamisme dont j'ai fait preuve. Afin de accroître mes compétences et d'assimiler le maximum de savoir possible, j'ai su rester à l'écoute de mes collègues qui m'ont toujours soutenus.

6. Lexique

- (1) **Covadis** : Applicatif d'AutoCAD en topographie, cartographie, terrassement, infrastructure et projets VRD.
- (2) **Blocs** : Terme générique désignant un ou plusieurs objets combinés pour créer un objet unique.
- (3) **Polylignes** : Objet composé d'un ou de plusieurs segments de droite ou arcs reliés entre eux, et considérés comme un seul objet. Une polyligne est considérée comme une seule ligne.
- (4) **Ortho-photographie** : Document photographique obtenu par redressement, mise à l'échelle et assemblage des surfaces élémentaires d'une photographie aérienne.
- (5) **Carte IGN** : Représentation à moyenne ou petite échelle des éléments naturels et artificiels situés sur la surface terrestre, ainsi que des formes du terrain.
- (6) **Attributs** : Entité textuelle définie par l'utilisateur utilisée pour le stockage de données. Un attribut de bloc est un membre imbriqué d'une définition de bloc AutoCAD et peut être configuré pour être visible ou invisible.
- (7) **Profil en travers** : Représentation par une coupe verticale perpendiculaire de l'axe (*profil en long*) du projet
- (8) **Profil en long** : Représentation d'une coupe verticale suivant l'axe d'un projet linéaire.
- (9) **Plan de comparaison** : C'est l'axe des abscisses du graphique sur lequel sont reportées les distances horizontales suivant l'axe du projet. (*Plan horizontal de référence*).
- (10) **Calque** : Regroupement logique de données comparable à des calques superposables sur un dessin. Vous pouvez visualiser les calques séparément ou en combinaison.
- (11) **SCU** : Système de coordonnées défini par l'utilisateur qui spécifie l'orientation des axes X, Y et Z dans un espace 3D. Le SCU détermine la position par défaut des objets dans un dessin.
- (12) **Ligne de commande** : Zone de texte réservée aux entrées clavier, aux invites et aux messages.
- (13) **Routine** : Partie du code d'un programme informatique destiné à être utilisé plus d'une fois.
- (14) **Invites** : Message sur la ligne de commande ou dans une info-bulle invitant l'utilisateur à indiquer des informations ou à exécuter une action, telle que la spécification d'un point.
- (15) **Geler** : Paramètre supprimant l'affichage des objets sur les calques sélectionnés. Les objets se trouvant sur les calques gelés ne sont ni affichés, ni régénérés, ni tracés.
- (16) **Clichés** : Vues d'espace objet dans un dessin, enregistré par Autocad. Ces images peuvent apparaître dans les boîtes de dialogue.
- (17) **Espace objet** : Un des deux espaces principaux où les objets résident. En règle générale, le modèle géométrique est défini dans un système de coordonnées en 3D appelé espace objet. Une présentation finale des vues et des annotations spécifiques à ce modèle est placée dans l'espace papier.
- (18) **Rendre courant** : Définit le calque sélectionné en tant que calque courant. Les objets que vous créez sont dessinés sur le calque courant.

7. Annexes

Annexe 1 : **PA1 - Plan de situation du terrain 1/10000^{ème}**

Annexe 2 : **PA2 – Notice décrivant le terrain et le projet d'aménagement prévu**

Annexe 3 : **PA3 - Plan de l'état actuel du terrain à aménager et de ses abords 1/3000^{ème} (A3)**

Annexe 4 : **PA4 - Plan de composition d'ensemble du projet coté dans les trois dimensions 1/2000^{ème} (A3)**

Annexe 5 : **Formulaire CERFA Permis d'Aménager SAGNES GENTIANES**

Annexe 6 : **Plan de situation 1/25000^{ème}**

Annexe 7 : **Plan cadastrale déboisement 1/125^{ème}**

Annexe 8 : **Plan reboisement 1/2000^{ème}**

Annexe 9 : **Demande autorisation de défrichement - cle075e1b - SAGNES GENTIANES**

Annexe 10 : **Programme LISP (LISP + DCL)**