

Mode d'emploi

# MUSLIW

## Logiciel de calcul d'accessibilité multimodale

### Sommaire

<a href="#">1 - Présentation de MUSLIW.....</a>	<a href="#">5</a>
<a href="#">2 - Eléments théoriques.....</a>	<a href="#">8</a>
<a href="#">3 - L'accessibilité en transports individuels (plus conseillée).....</a>	<a href="#">14</a>
<a href="#">4 - Convertir un réseau "CHOQUETTE".....</a>	<a href="#">21</a>
<a href="#">5 - Convertir un réseau GTFS (Google Transit Feed Specification).....</a>	<a href="#">22</a>
<a href="#">6 - L'accessibilité en transports collectifs à horaires.....</a>	<a href="#">23</a>
<a href="#">7 - Exemples d'utilisation.....</a>	<a href="#">46</a>
<a href="#">8 - Simplification d'un réseau.....</a>	<a href="#">52</a>
<a href="#">9 - Visualisation.....</a>	<a href="#">52</a>
<a href="#">10 - Exploiter les résultats.....</a>	<a href="#">57</a>
<a href="#">11 - Glossaire.....</a>	<a href="#">61</a>
<a href="#">Bibliographie.....</a>	<a href="#">62</a>

Les études d'accessibilité font appel à un nombre important de données qu'il faut traiter informatiquement avec des outils adaptés. Les logiciels de planification des transports proposent de tels outils mais ils peuvent être onéreux, parfois difficiles à mettre en œuvre et nécessitent généralement un investissement humain important (prise en main du logiciel, codification des données, etc.). Par ailleurs, les algorithmes permettant le calcul d'accessibilité sont propres à chaque logiciel et peuvent être inadaptés à certains cas (accessibilité multimodale, prise en compte de fiches horaires dans les TC, codage de systèmes de vélo en libre service, temps de parcours routiers variables selon l'heure et le jour, etc.).

Afin de permettre des calculs d'accessibilité multimodale à partir de données avec des formats largement répandus, le CETE Nord-Picardie a développé un utilitaire appelé MUSLIW.

Ce document décrit comment utiliser l'outil MUSLIW. Il présente les fondements théoriques de la recherche de plus court chemin sur un graphe et les principes retenus pour les algorithmes implémentés dans le logiciel. Il détaille le fonctionnement de l'utilitaire pour calculer des indicateurs d'accessibilité sur des réseaux multimodaux mêlant transport individuel et transport collectif. Il fournit également quelques exemples de codification et de paramétrage permettant de répondre à plusieurs types de questions liées à l'accessibilité. Enfin, il décrit les autres fonctionnalités de l'outil MUSLIW : simplification de réseau, affichage de cartes et de données.

Page laissée blanche intentionnellement

Mode d'emploi

# MUSLIW



Logiciel de calcul d'accessibilité  
multimodale

# Introduction

Les études d'accessibilité font appel à un nombre important de données qu'il faut traiter informatiquement avec des outils adaptés. Les logiciels de planification des transports proposent de tels outils mais ils peuvent être onéreux, parfois difficiles à mettre en œuvre et nécessitent généralement un investissement humain important (prise en main du logiciel, codification des données, etc.). Par ailleurs, les algorithmes permettant le calcul d'accessibilité sont propres à chaque logiciel et peuvent être inadaptés à certains cas (accessibilité multimodale, prise en compte de fiches horaires dans les TC, codage de systèmes de vélo en libre service, temps de parcours routiers variables selon l'heure et le jour, etc.).

Afin de permettre des calculs d'accessibilité multimodale à partir de données avec des formats largement répandus, le CETE Nord-Picardie a développé un utilitaire appelé MUSLIW.

Ce document décrit comment utiliser l'outil MUSLIW. Il présente les fondements théoriques de la recherche de plus court chemin sur un graphe et les principes retenus pour les algorithmes implémentés dans le logiciel. Il détaille le fonctionnement de l'utilitaire pour calculer des indicateurs d'accessibilité sur des réseaux multimodaux mêlant transport individuel et transport collectif. Il fournit également quelques exemples de codification et de paramétrage permettant de répondre à plusieurs types de questions liées à l'accessibilité. Enfin, il décrit les autres fonctionnalités de l'outil MUSLIW : simplification de réseau, affichage de cartes et de données.

# 1 - Présentation de MUSLIW

## 1.1 - Historique

### 1.1.1 - Premiers travaux

En 1995, le DRE Picardie demanda au CETE Nord-Picardie de réaliser pour le compte de l'Observatoire Régional des Transports, une étude sur la production de cartes et d'indicateurs d'accessibilité routière et en transports collectifs.

A l'époque, les SIG (Systèmes d'Informations Géographiques) commençaient seulement à se démocratiser et à se diffuser. Le Ministère s'orientant plutôt vers le choix de MapInfo, le CETE Nord-Picardie acquit ce logiciel, qui ne dispose pas en standard de module de type cheminement sur un graphe.

Le CETE Nord-Picardie a dans un premier temps développé un algorithme de plus court chemin sous la forme d'un exécutable autonome écrit en C++ (baptisé chemins.exe) et un utilitaire MapBasic pour construire la topologie du réseau (numérotation des nœuds notamment).

Cet algorithme (Dijkstra) basé sur une typologie de type «nœud» permettait également de calculer l'accessibilité en transports collectifs avec rabattement en voiture. Les cartes résultant de ces calculs étaient établies pour la période de pointe du matin (arrivée au pôle entre 7h et 9h). La première étape consistait en un calcul du temps d'accès au pôle étudié, depuis chaque gare (temps de parcours du train le plus rapide reliant la gare et le pôle avec prise en compte des correspondances). Dans un second temps, pour toutes les autres communes, un calcul de plus court chemin, combinant accès en voiture vers une gare et trajet en train de la gare au pôle était effectué. Un temps de correspondance de 10 minutes était introduit. Pour tenir compte du coût du déplacement, le temps en voiture était doublé.

### 1.1.2 - Principales évolutions

L'algorithme de plus court chemin a ensuite été intégré dans une bibliothèque dynamique (DLL). Cette amélioration permettait de réaliser toutes les étapes d'un calcul d'accessibilité et de sa représentation depuis Mapinfo :

- génération du graphe ;
- calcul d'accessibilité ;
- dessin des isochrones.

L'algorithme de plus court chemin a également intégré la possibilité de calcul selon une topologie de type «noeud» ou «tronçon» (cf. § 2.1. pour une présentation de ces concepts) permettant d'introduire des interdictions ou des pénalités de mouvements tournants.

L'outil a enfin été enrichi d'un algorithme de plus court chemin de type «stratégies optimales» pour les réseaux de transports collectifs décrits selon la fréquence des lignes, similaire à celui proposé dans Emme/2 et d'une amélioration du dessin des isochrones tenant compte des coupures urbaines et de la continuité des temps de parcours sur les tronçons, au lieu des temps aux nœuds comme précédemment. L'algorithme de recherche de chemins en stratégies optimales fonctionne toujours dans l'outil MapInfo (RESEAUX.MBX,

développé par le CETE Nord-Picardie) même s'il aurait besoin d'être testé plus en profondeur pour vérifier son absence de bugs.

L'outil a même essayé de proposer un calcul d'accessibilité basé sur les horaires, mais l'algorithme utilisé précédemment n'était pas assez performant pour effectuer un tel calcul dans des délais raisonnables. La relative ancienneté de l'outil de programmation n'autorisait pas l'utilisation aisée d'objets containers (liste de listes, queues, etc.), ce qui ne permettait plus à l'outil d'évoluer.

### **1.1.3 - Développement de MUSLIW**

C'est pour contourner ces limitations qu'a été développé MUSLIW sous Visual C# Express 2005. L'intérêt était de pouvoir construire, à l'aide d'objets containers performants, un réseau TC dans un modèle de données parfaitement adapté à l'algorithme de calcul de plus court chemin et d'améliorer les performances de l'algorithme en lui-même.

MUSLIW permet notamment de calculer les plus courts chemins en temps généralisé dans un réseau multimodal en tenant compte des horaires et du calendrier des services et d'affecter la demande correspondante en choisissant un jour et une heure soit de départ, soit d'arrivée. MUSLIW permet également de tenir compte de temps de pénalités de mouvement tournant ou de temps de correspondance spécifiques définis par l'utilisateur ainsi que des temps de parcours routiers en fonction de l'heure et du calendrier, de l'ouverture à la circulation sur une période restreinte de la journée et/ou à certains jours seulement.

MUSLIW est un outil interne au CETE Nord-Picardie. Ses fonctionnalités sont développées au fur et à mesure des besoins de celui-ci et ne sont parfois pas encore complètement mises en œuvre. Pour la maintenance du logiciel, il convient de s'adresser au CETE Nord-Picardie directement.

## **1.2 - Installation**

### **1.2.1 -Sous Windows XP**

MUSLIW fonctionne sous Windows XP et nécessite l'installation du Microsoft .NET Framework 2.0, téléchargeable gratuitement sur le site de Microsoft.

L'outil est diffusé sous la forme d'un fichier ZIP. Pour l'installer, il suffit de faire les manipulations suivantes :

- installer au préalable si nécessaire Microsoft .NET Framework 2.0 ;
- décompresser le fichier MUSLIW.ZIP dans un répertoire ;
- lancer le programme SETUP.EXE.

Le Programme se lance ensuite via Démarrer/Programmes/CETE Nord-Picardie/MUSLIW. L'image ci-contre fournit un aperçu de l'interface de l'outil.

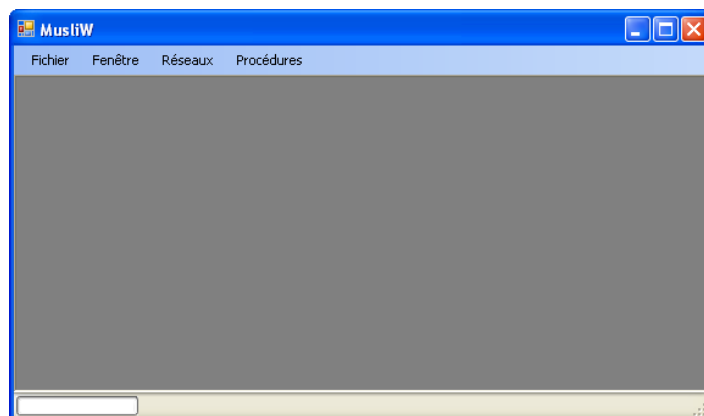


Figure 1: Fenêtre principale de l'outil sous Windows XP

### 1.2.2 -Sous Linux

Une version de test existe également sous Linux. Elle fonctionne avec Mono 2.0. L'outil a été testé sous la distribution Ubuntu 9.04 Jaunty et 9.10 Karmic. L'affichage des tableaux de données n'est par contre pas opérationnel.

L'outil est diffusé sous la forme d'une archive au format ZIP contenant 2 fichiers :

- MUSLIW : le lanceur qui permet de lancer Musliw.exe avec Mono.
- MUSLIW.exe : l'exécutable qui fonctionne également sous Windows.

Pour l'installer, il suffit de faire les manipulations suivantes :

- extraire tous les fichiers dans un répertoire ;
- autoriser le lanceur à être exécuté
- exécuter le lanceur



Figure 2 : Ecran principal sous Linux

## 2 - Éléments théoriques

L'outil MUSLIW intègre aujourd'hui différents algorithmes de recherche de chemins sur un graphe. Afin de les présenter, cette partie explique les principaux concepts de topologie des réseaux nécessaires puis détaille ces algorithmes en précisant comment les paramétrer au sein de l'outil.

### 2.1 - Construction de la topologie de réseau

Pour pouvoir cheminer dans un graphe, il est nécessaire d'adopter des règles de cheminement sur ce dernier, c'est-à-dire une «topologie de réseau».

La construction d'un graphe de réseau consiste à créer des nœuds, avec une numérotation, auxquels sont connectés tous les segments. Ainsi, on sait que chaque segment  $i,j$  commence au nœud «i» et se termine au nœud «j».

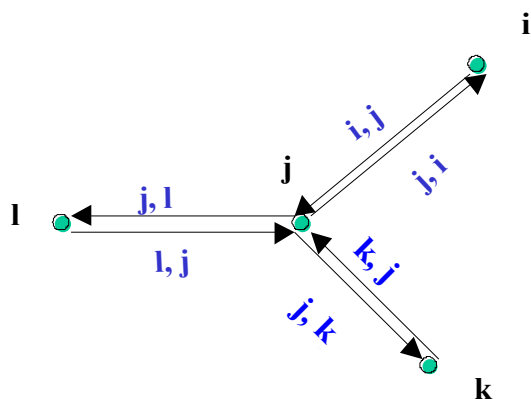


Figure 3 : Graphe de réseau simple

Une fois ce graphe construit, il est possible de cheminer sur celui-ci en définissant différentes topologies :

- de type nœud ;
- de type tronçon ;
- de type mouvement tournant.

#### 2.1.1 -Topologie de type nœud

Pour cheminer sur le graphe, la manière la plus simple est de se déplacer de nœud en nœud. Chaque nœud du graphe possède des nœuds prédécesseurs et successeurs. On chemine d'un nœud à un autre par l'intermédiaire d'un tronçon orienté reliant les deux nœuds.

Ce type de problématique est souvent traité avec des algorithmes de type Dijkstra qui permettent de déterminer les coûts minimaux pour atteindre chaque nœud par rapport à un nœud origine.



Le coût de chaque nœud du réseau est successivement minimisé à partir de celui des nœuds prédécesseurs par l'intermédiaire de l'arc reliant les deux nœuds :

$$P_j = P_i + C(ij)$$

Où :

- $P_j$  coût au nœud  $j$  ;
- $P_i$  coût au nœud  $i$  ;
- $C(ij)$  coût du tronçon  $i,j$ .

Cette topologie offre la possibilité de déterminer les coûts uniquement aux nœuds et non en n'importe quel point du réseau. Elle ne permet pas directement la prise en compte de pénalités ou d'interdictions de mouvements tournants, pour cela il est nécessaire d'éclater virtuellement les nœuds concernés.

### 2.1.2 -Topologie de type tronçon

Dans ce type d'approche, le cheminement ne se fait plus de nœud en nœud mais de tronçon en tronçon. Ainsi, chaque tronçon possède des tronçons prédécesseurs et successeurs.

Dans le cas de la topologie de type nœud, la longueur parcourue par le plus court chemin pour aller aux nœuds  $i, j, k, l$ , est connue, mais il est impossible de connaître la longueur parcourue au point situé à mi-chemin entre  $i$  et  $j$  par exemple.

Dans cette approche de type tronçon, le coût de chaque nœud du réseau n'est plus minimisé à partir de celui des nœuds prédécesseurs par l'intermédiaire des arcs les reliant.

La relation est en effet la suivante :

Le coût de chaque tronçon du réseau est successivement minimisé à partir de celui des tronçons prédécesseurs par l'intermédiaire d'un mouvement tournant reliant les deux tronçons :

$$P_{jk} = P_{ij} + T(ijk)$$

Où :

- $P_{jk}$  coût au tronçon  $j,k$  ;
- $P_{ij}$  coût au tronçon  $i,j$  ;
- $T(ijk)$  coût du mouvement tournant  $i, j, k$ .

La relation ci-dessus confirme que ce type de topologie traite en standard la problématique des mouvements tournants.

Le coût d'un tronçon est alors défini par convention, soit comme le coût au nœud extrémité, soit au nœud origine.

Pour un tronçon  $i, j$  :

$$\text{Coût (au nœud } j \text{ du tronçon } i, j) = \text{Coût (au nœud } i \text{ du tronçon } i, j) + \text{Coût (tronçon } i, j)$$

$$P_{ij}^j = P_{ij}^i + C(ij)$$

Où :

- $P_{ij}^i$  coût au nœud i du tronçon i, j ;
- $P_{ij}^j$  coût au nœud j du tronçon i, j ;
- $C(ij)$  coût du tronçon tournant i, j.

Cela implique que le coût à un nœud n'est pas unique. Sur l'exemple du schéma ci-dessus, le coût du nœud J est défini par les coûts des tronçons l, j ; k, j et i, j :

$$P_j = \text{Min} (P_{ij} + P_{kj} + P_{lj})$$

Où :

- $P_{kj}$  coût au tronçon k, j ;
- $P_{ij}$  coût au tronçon i, j ;
- $P_{lj}$  coût au tronçon l, j.

Il est également possible de connaître la longueur parcourue en n'importe quel point du réseau et non uniquement aux extrémités des segments.

Soit M un point sur le tronçon i, j. On définit  $L_{im}$ , longueur partielle du tronçon entre i et m telle que :

$$L_{im} = \lambda L_{ij}, \text{ avec } \lambda \in [0,1]$$

Alors :

$$P_m = P_{ij} - \lambda C(ij)$$

Où :

- $P_m$  coût au point m ;
- $P_{ij}$  coût au tronçon i, j ;
- $C(ij)$  coût du tronçon i, j.

Cette formulation permet d'introduire et de traiter en standard et de manière systématique les interdictions et les pénalités de mouvements tournants.

En revanche, si le traitement systématique des phénomènes d'entrecroisement n'est pas possible avec cette topologie (cf. infra), elle peut être mise en œuvre en éclatant virtuellement les tronçons, mais cette fonctionnalité n'est pas disponible dans la version actuelle du logiciel.

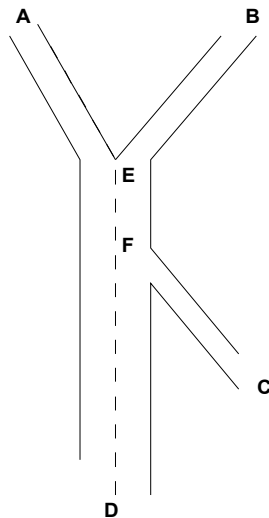


Figure 4 : Schéma d'entrecroisement

Un entrecroisement est défini par une succession de trois tronçons.

Sur le schéma ci-dessus qui représente un sens d'autoroute à 2 voies, il est possible que les véhicules entrant en C aient l'interdiction d'aller en A parce qu'ils doivent changer de file sur une très courte distance, alors que ceux en section courante venant de D ont le droit de bifurquer soit vers A ou vers B.

La topologie de tronçon ne permet pas en standard de spécifier de telles interdictions.

### 2.1.3 -Topologie de type mouvement tournant

Dans cette approche, le cheminement se fait de mouvement tournant en mouvement tournant. Ainsi chaque mouvement tournant dispose de mouvements tournants successeurs et prédécesseurs.

Cette topologie a l'avantage de traiter en standard et de manière systématique les problématiques d'entrecroisements complexes pour lesquelles l'autorisation de tourner dépend du mouvement par lequel l'usager s'est inséré sur le tronçon. Elle permet, par exemple, de spécifier le cas présenté au schéma précédent.

Cette topologie a l'inconvénient de rendre la construction du graphe assez complexe et peu intuitive. Cette notion de complexité dans la topologie peut être généralisée à des portions d'itinéraires.

Dans MUSLIW c'est la topologie de type tronçon qui a été mise en œuvre :

- depuis la version 1.2, MUSLIW permet de traiter les pénalités et interdictions de mouvements tournants et de correspondance ;
- il n'est en conséquence pas possible de traiter les problèmes d'entrecroisement.

## 2.2 - Algorithme utilisé

### 2.2.1 -Description

Des travaux de recherche montrent que deux algorithmes sont particulièrement performants notamment sur les gros réseaux routiers<sup>1</sup> :

- *Graph Growth Algorithm implemented with two queues* ;
- *Dijkstra with approximate buckets*.

Le calcul de plus court chemin routier utilise le *Graph Growth Algorithm with two queues*.

Pour le calcul sur un réseau TC à horaires et selon le calendrier connecté à un réseau de transport individuel, cet algorithme s'avère ne pas être très performant de par la présence de tronçons TC dont l'heure de départ du service est très postérieure à l'heure de départ souhaitée.

L'algorithme de *Dijkstra with approximate buckets*, bien plus performant pour ce type de réseau, est disponible dans MUSLIW. Cependant, c'est en mixant les deux algorithmes, pour constituer le *Graph Growth Algorithm with approximate buckets*, que l'on semble obtenir la meilleure performance.

Le cœur de l'algorithme est le *Graph Growth Algorithm* à la différence près que les éléments sont rangés par intervalles et non stockés dans deux queues. La performance de l'algorithme est dépendante de la manière de construire les différents intervalles. Il est toutefois plus difficile à paramétrer.

Pour une performance optimale, la file d'attente des tronçons marqués devrait être triée par ordre croissant du coût généralisé, mais les procédures de tri coûtent cher en temps de calcul. L'idée est donc de les stocker dans des intervalles en fonction de leur coût généralisé. Ainsi les premiers qui seront analysés seront ceux au coût le plus faible. Cela minimise la probabilité qu'ils soient remis dans la file d'attente car un itinéraire plus rapide aura été trouvé.

### 2.2.2 -Paramètres

Les paramètres de l'algorithme sont les intervalles et la façon dont ils sont définis.

La manière la plus simple de définir les intervalles est de découper l'amplitude des coûts généralisés qui seront calculés dans la recherche de plus court chemin en intervalles de même amplitude.

Cela nécessite déjà de déterminer une amplitude théorique et un nombre d'intervalles.

- **Max classes** : définit le nombre d'intervalles utilisés par l'algorithme ;
- **paramètre d'amplitude** : permet de fixer l'amplitude des temps généralisés.

Toutefois, les réseaux routiers ou transports collectifs sont décrits généralement dans un espace à 2 dimensions.

---

<sup>1</sup> **Zhan, F. B., and Noon, C. E.** (1996) Shortest Path Algorithms : An Evaluation Using Real Road Networks. Transportation Science.

Sous la condition que les nœuds et les tronçons soient uniformément répartis dans l'espace, à partir du point d'origine du calcul de plus court chemin, si  $n$  objets sont recensés dans un disque de rayon  $R$ , alors le disque de rayon  $R \cdot x$  ( $x > 0$ ) contient potentiellement  $n \cdot x^2$  objets. Le nombre potentiel d'objets dans un intervalle croît donc a priori en fonction d'une loi puissance deux, pour les réseaux surfaciques, si l'on considère des intervalles d'amplitude égale.

Or, la performance de l'algorithme est meilleure avec des intervalles ayant le même nombre d'éléments plutôt que des intervalles de même amplitude. L'algorithme fait en sorte de définir des intervalles afin que le nombre d'éléments dans ces intervalles soit identique. On définit ensuite ces intervalles de la façon suivante :

Soit  $i$  l'intervalle où sera stocké le coût intermédiaire de chaque tronçon dans la phase de calcul de plus court chemin :

$$i = \min \left( E \left( \frac{t^\alpha}{p}, n \right) \right)$$

Où :

- $n$  = nombre maximal d'intervalles (max classes) ;
- $p$  = paramètre d'amplitude (paramètre) ;
- $t$  = coût intermédiaire du tronçon en minutes ;
- $\alpha$  = paramètre puissance (2 en général pour les réseaux surfaciques) ;
- $E(x)$  : partie entière de  $x$ .

Ainsi, avec les paramètres proposés par défaut (max classes = 10000 et paramètre = 100), un tronçon de temps généralisé de 10mn sera stocké dans le premier intervalle (limite avec le 2<sup>ème</sup>) alors qu'un tronçon de temps généralisé de 20mn sera dans le 4<sup>ème</sup> (limite avec le 5<sup>ème</sup>).

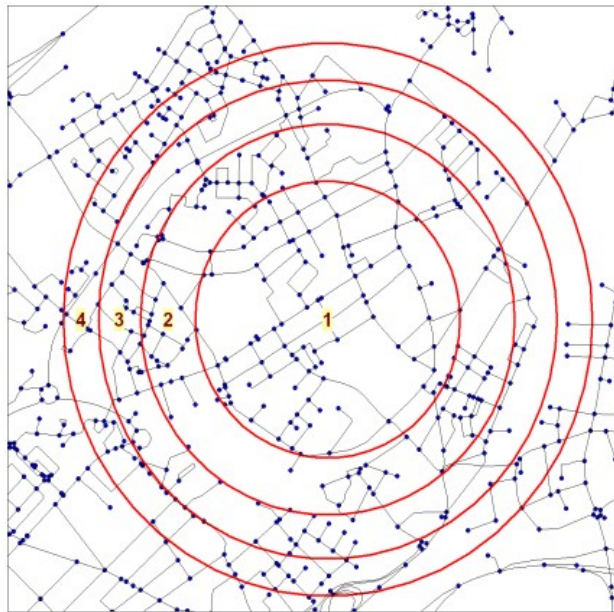


Figure 5 : Paramétrage de l'algorithme

L'exemple ci-contre permet d'illustrer le principe du paramétrage :

- le "paramètre d'amplitude" détermine la taille du 1<sup>er</sup> cercle ;
- "Max classes" détermine le nombre maximum de classes.

On considère une puissance  $\alpha = 2$  par défaut.

Alors, le nombre de nœuds contenus dans chaque classe est sensiblement égal :

- classe 1 : 94 ;
- classe 2 : 104 ;
- classe 3 : 98 ;
- classe 4 : 106.

## 3 - L'accessibilité en transports individuels (plus conseillée)

### 3.1 - Introduction

Le calcul de l'accessibilité en transports individuels s'effectue en plusieurs étapes. Il faut :

- importer un ou plusieurs réseaux dans le logiciel ;
- charger une ou plusieurs matrices dans le logiciel ;
- effectuer une affectation en tout ou rien sur un de ces réseaux avec une de ces matrices.

Les résultats de l'affectation donnent alors les temps de parcours sur chaque origine-destination non nulle de la matrice. Ces informations permettent au final de réaliser des analyses d'accessibilité.

Cet ancien module avait été mis en œuvre afin de pouvoir calculer des matrices de temps de parcours avec une rapidité optimale. Aujourd'hui, l'affectation TC à horaires permet de réaliser la même chose dans des temps de calcul équivalents et d'effectuer des analyses en sortie beaucoup plus poussées. Elle permet aussi la prise en compte des pénalités ou des interdictions de mouvements tournants et un paramétrage des temps de parcours variables dans le temps et selon les jours. **L'utilisation de l'affectation TC à horaires est fortement recommandée.**

Dans cette procédure, les matrices stockées en mémoire et l'absence d'horaire permettent de traiter toutes les destinations d'une origine en une seule opération, ce qui est bien plus efficace quand la matrice comporte de nombreuses cases non nulles et que l'on souhaite mesurer l'accessibilité d'un nœud à un grand ensemble d'autres nœuds du réseau.

L'algorithme utilisé est le «*Graph Growth Algorithm à intervalles*». Les paramètres sont ceux proposés par défaut dans le calcul à horaires.

## 3.2 - Importer un réseau de transports individuels MUSLIW

Le format du réseau doit être un fichier de type texte avec un « » (espace) comme séparateur et avec comme colonnes, de gauche à droite :

- nœud début : numéro de nœud d'origine du tronçon ;
- nœud fin : numéro de nœud de fin du tronçon ;
- temps : temps de parcours ou coût. C'est ce champ que l'algorithme tentera de minimiser dans le calcul des plus courts chemins ;
- distance : variable qui sera cumulée le long de l'itinéraire au coût minimal.

Voici ci-après un exemple de réseau de transport individuel défini pour réaliser un calcul d'accessibilité avec MUSLIW.

L'import s'effectue grâce au menu Fichier/Importer réseau accessibilité. C'est le même format de fichier que celui utilisé dans l'utilitaire Mapinfo RESEAUX.MBX. Cet outil RESEAUX.MBX traite les mouvements tournants et a été développé par le Cété Nord-Picardie.

```
15888 15648 0.04737 0.18947
15888 15775 0.01852 0.07407
15888 15950 0.02455 0.09820
15889 15818 0.01130 0.04521
15889 15858 0.00589 0.02356
15889 15922 0.00944 0.03775
15889 15632 0.18974 0.75898
15890 16026 0.02242 0.08969
15891 15874 0.01668 0.06670
15891 15893 0.03083 0.12330
15891 16222 0.06107 0.24429
15892 15864 0.00932 0.03728
15893 15869 0.00779 0.03114
15893 15891 0.03083 0.12330
15893 15968 0.01148 0.04590
15894 15845 0.01187 0.04749
15895 15841 0.01147 0.04588
15895 15852 0.01177 0.04707
15895 15928 0.01105 0.04422
15895 15966 0.01712 0.06848
15896 15871 0.00361 0.01444
15896 15926 0.02772 0.11089
15896 16131 0.04219 0.16876
15897 15740 0.02678 0.10711
15897 15834 0.01460 0.05840
15897 15946 0.00848 0.03394
15897 16210 0.08130 0.32519
15898 15709 0.02740 0.10959
15899 15873 0.01146 0.04584
15899 15944 0.01619 0.06477
```

### 3.3 - Importer un réseau Emme/2

MUSLIW prévoit d'importer directement un réseau du type de ceux définis dans Emme/2.

Pour ces réseaux Emme/2, le format du fichier est le suivant :

c EMME/2 Module: 2.14(v9.03) Date: 06-11-08 11:07 User: E118/LILLEART...am

c Project: essai

c Scenario 1: reseau vp dunkerque

t nodes

a*	50010	583835	366863	5	50010	9	0000
a*	50021	584493	367098	5	50021	9	0000
a*	50022	586670	368681	5	50022	9	0000
a*	50031	585218	365378	5	50031	9	0000
a*	50032	586633	366146	5	50032	9	0000
a*	50033	586378	365083	5	50033	9	0000
a*	50034	585060	364518	5	50034	9	0000
a*	50050	590149	360917	5	50050	9	0000
a*	50051	590210	361401	5	50051	9	0000
a*	50052	590696	360800	5	50052	9	0000
a*	50053	590146	360107	5	50053	9	0000
a*	50054	589019	358833	5	50054	9	0000
a*	50055	587929	361236	5	50055	9	0000
a*	50056	592100	362605	5	50056	9	0000
a*	50057	593023	360249	5	50057	9	0000
a*	50060	587170	362938	5	50060	9	0000
a*	50070	592579	363419	5	50070	9	0000
a*	50071	591950	366458	5	50071	9	0000
a*	50072	591667	365785	5	50072	9	0000

t links

a	50010	51066	0.10	vpb	9	9.9	11	5	0	9
a	50010	51120	0.10	vpb	9	9.9	11	5	0	9
a	50010	51168	0.10	vpb	9	9.9	11	5	0	9
a	50021	51287	0.10	vpb	9	9.9	11	5	0	9
a	50021	51305	0.10	vpb	9	9.9	11	5	0	9
a	50022	51566	0.10	vpb	9	9.9	11	5	0	9
a	50022	51630	0.10	vpb	9	9.9	11	5	0	9
a	50031	51110	0.10	vpb	9	9.9	11	5	0	9
a	50031	51406	0.10	vpb	9	9.9	11	5	0	9
a	50031	51428	0.10	vpb	9	9.9	11	5	0	9
a	50032	51493	0.10	vpb	9	9.9	11	5	0	9
a	50032	51541	0.10	vpb	9	9.9	11	5	0	9
a	50033	51396	0.10	vpb	9	9.9	11	5	0	9
a	50033	51447	0.10	vpb	9	9.9	11	5	0	9
a	50034	51079	0.10	vpb	9	9.9	11	5	0	9
a	50034	51392	0.10	vpb	9	9.9	11	5	0	9
a	50050	51747	0.10	vpb	9	9.9	11	5	0	9
a	50050	51769	0.10	vpb	9	9.9	11	5	0	9
a	50051	51816	0.10	vpb	9	9.9	11	5	0	9



### Description de la syntaxe :

Les lignes qui commencent par «c» sont des commentaires facultatifs.

La ligne t nodes précède les enregistrements des nœuds et centroïdes. Le format est de type texte délimité par des blancs, les colonnes renvoient, dans l'ordre aux informations suivantes :

- a : signifie ajouter (ici, un nœud) et \* précise que ce nœud est un centroïde ;
- i : numéro du nœud ou centroïde ;
- x : abscisse du nœud ;
- y : ordonnée du nœud ;
- ui1 : 1ère donnée utilisateur ;
- ui2 : 2ème donnée utilisateur ;
- ui3 : 3ème donnée utilisateur.

Après les données de nœuds, la ligne t links précède les données de tronçons. Le format est de type texte délimité par des blancs, les colonnes renvoient, dans l'ordre aux informations suivantes :

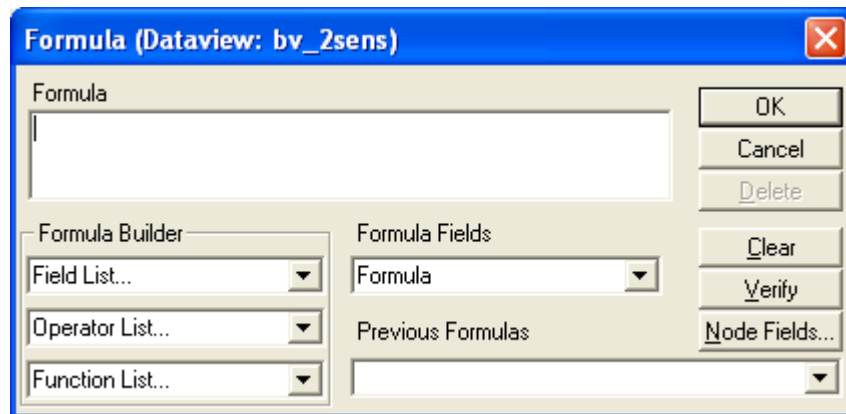
- a : ajouter un tronçon ;
- i : numéro de nœud du début du tronçon ;
- j : numéro de nœud de fin du tronçon ;
- len : longueur du tronçon ;
- modes : chaîne de caractères décrivant les modes autorisés à circuler sur les tronçons ;
- type : type du tronçon ;
- lan : nombre de voies du tronçon (variable déterminant la capacité) ;
- vdf : numéro de fonction temps de parcours/volume ;
- ul1 : donnée utilisateur numéro1 ;
- ul2 : donnée utilisateur numéro2 ;
- ul3 : donnée utilisateur numéro3.

L'importation se fait via le menu «Fichier>Importer Réseau Emme/2».

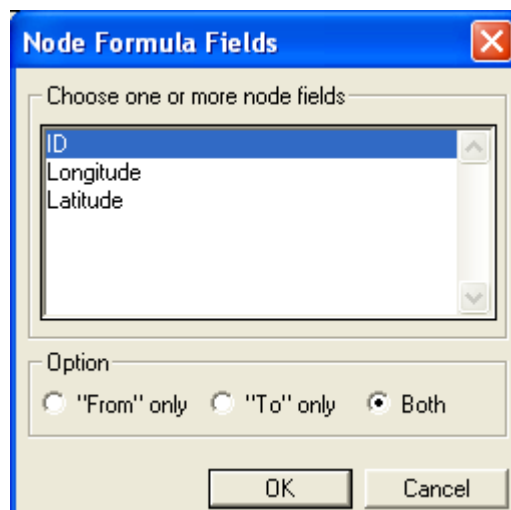
## 3.4 - Importer un réseau TransCAD

Il n'est pas possible d'importer directement un réseau TransCAD dans MUSLIW. Toutefois, TransCAD peut être d'une très grande utilité pour générer les numéros de nœuds d'une couche géographique d'objets linéaires. Il suffit pour cela de réaliser la procédure suivante :

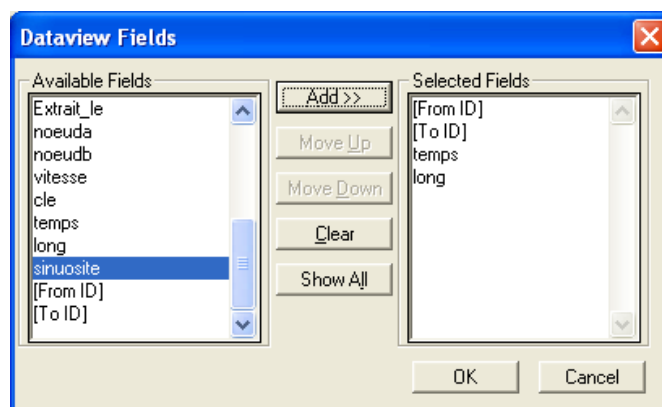
- afficher un *DataView* de votre table réseau ;
- afficher les numéros de nœuds avec «*Dataview>Formula Fields*» ;



- cliquer sur «Node Fields...» ;



- sélectionnez «ID» et «Both» ;
- sélectionnez les champs nécessaires à la création du réseau via la commande «Dataview>Fields» en n'affichant que ceux nécessaires (Cela suppose d'avoir renseigné chaque arc du réseau avec un temps de parcours) ;



- [From ID] : numéro de nœud initial ;
- [To ID] : numéro de nœud final ;
- temps : un champ temps ;
- long : un champ longueur. Exportez la *DataView* sous forme de fichier dBase, ou texte.

Attention, TransCAD ne permet l'export au format texte délimité qu'avec le séparateur « , ». Or MUSLIW nécessite comme délimiteur :

- « » pour les réseaux de transport individuel ;
- « ; » pour les réseaux multimodaux.

Une manipulation supplémentaire sera nécessaire pour transformer « , » en « ; » ou en « ».

Pour tenir compte des sens de circulation, vous devez effectuer deux sélections :

- sélectionner les éléments pour lesquels DIR vaut 0 ou 1 (double sens ou sens du flux égal au sens topologique) et afficher les colonnes dans l'ordre suivant :
  - [From ID] : numéro de nœud initial ;
  - [To ID] : numéro de nœud final ;
  - un champ temps ;
  - un champ longueur.
- sélectionner les éléments pour lesquels DIR vaut 0 ou -1 (double sens ou sens du flux inverse au sens topologique) et afficher les colonnes dans l'ordre suivant :
  - [To ID] : numéro de nœud initial ;
  - [From ID] : numéro de nœud final ;
  - un champ temps ;
  - un champ longueur.

Il faut donc, dans ce deuxième cas, inverser l'ordre des numéros de nœud origine/destination et ensuite concaténer les deux fichiers texte exportés pour constituer le réseau MUSLIW.

## 3.5 - Importer une matrice

L'importation d'une matrice ne peut s'effectuer que lorsqu'au moins un réseau de transport individuel a déjà été importé. L'outil demande alors de choisir le réseau qui devra être mis en concordance avec la matrice stockée en mémoire.

Le fichier matrice doit avoir la forme suivante :

```
2 5 1
2 7 1
2 9 1
2 10 1
2 13 1
2 15 1
2 16 1
2 17 1
```

2 19 1.

Les trois colonnes des fichiers matrices correspondent dans l'ordre aux informations suivantes :

- origine : numéro de nœud origine du déplacement ;
- destination : numéro de nœud destination du déplacement ;
- volume : volume de trafic sur l'origine/destination.

**NB : Un tel fichier peut facilement être obtenu à partir des fichiers TransCAD de matrice.**

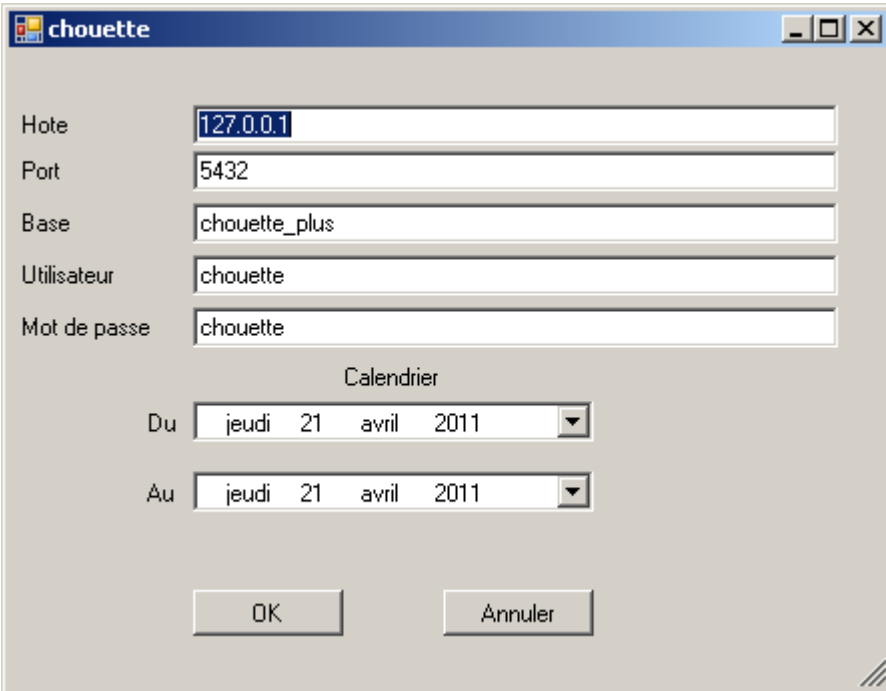
## 4 - Convertir un réseau "CHOUETTE"

CHOUETTE est un logiciel libre qui permet d'échanger des fichiers de référentiels d'offre de transports en commun. Il s'appuie notamment sur le format d'échanges normalisé Trident/Neptune.

Le logiciel fonctionne à partir d'une interface web, les données d'offre de transport étant stockées dans une base de données PostgreSQL.

Musliw dispose dorénavant d'une fonctionnalité qui permet de se connecter à la base PostgreSQL gérée par « Chouette » et de construire automatiquement un réseau utilisable directement sous Musliw pour les calculs de temps de parcours intermodaux.

Pour lancer la fonctionnalité, il faut exécuter le menu « Procédures->Convertir un réseau Chouette ». La fenêtre suivante apparaît.



Il suffit de saisir les paramètres de connexion à la base Chouette et de saisir le calendrier sur lequel on souhaite travailler (jour de début, jour de fin).

Musliw demandera alors le choix d'un nom de fichier text où il stockera le réseau converti.

Le réseau est alors directement exploitable avec Musliw pour le calcul d'itinéraires intermodaux.

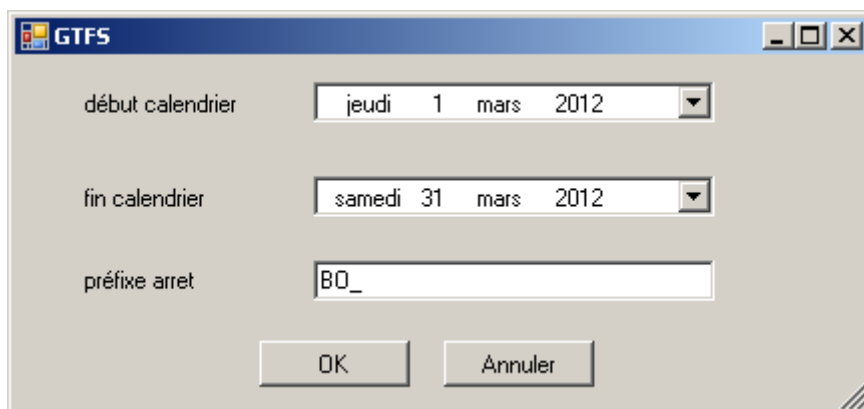
## 5 - Convertir un réseau GTFS (Google Transit Feed Specification)

Le développement de l'Open Data conduit à ce qu'un nombre de plus en plus important de collectivités locales mettent à disposition en libre téléchargement leur offre théorique de transports en commun.

Le format utilisé par les différentes collectivités est rarement le format d'échanges Neptune (normalisé), évolution du format Trident/Chouette, dont la conversion est déjà mis en œuvre dans Musliw, mais presque toujours le format GTFS de Google Transit qui s'impose de plus en plus comme un standard de fait.

Il est alors apparu pertinent que Musliw puisse convertir directement des fichiers GTFS au format Musliw.

Cette fonctionnalité s'active par le menu Procédure->Convertir Réseau GTFS (Google).



*Illustration 1: Dialogue pour la conversion GTFS*

La fenêtre ci-dessus présente les éléments nécessaires pour la conversion :

- La date de début du calendrier d'étude :
- La date de fin du calendrier d'étude
- Un préfixe (facultatif) devant les identifiant d'arrêts

Les dates de début et fin de calendrier permettent de construire un réseau Musliw sur une semaine par exemple ou un mois. Si les dates sont en dehors des limites existantes dans les données Google, les services seront considérés comme ne circulant pas ces jours là.

Seuls les services circulant au moins une fois entre les deux dates de calendrier d'étude seront convertis. Il est en effet inutile d'inclure dans le réseau Musliw des services qui ne pourront jamais être empruntés.

Le préfixe avant les identifiants d'arrêts peut s'avérer très utile quand il s'agit de combiner les réseaux de plusieurs autorités organisatrices qui peuvent disposer d'identifiants d'arrêts communs.

En cliquant sur Ok Musliw demandera de lui spécifier le répertoire où se situent les fichiers au format GTFS ainsi que de préciser ensuite le nom du fichier Musliw à créer

## 6 - L'accessibilité en transports collectifs à horaires

Pour le calcul d'accessibilité en transports collectifs à horaires, il faut importer dans l'outil un réseau et une matrice de demande. Il faut également paramétrer une boîte de dialogue pour le calcul du temps généralisé TC. Les paragraphes suivants présentent la démarche.

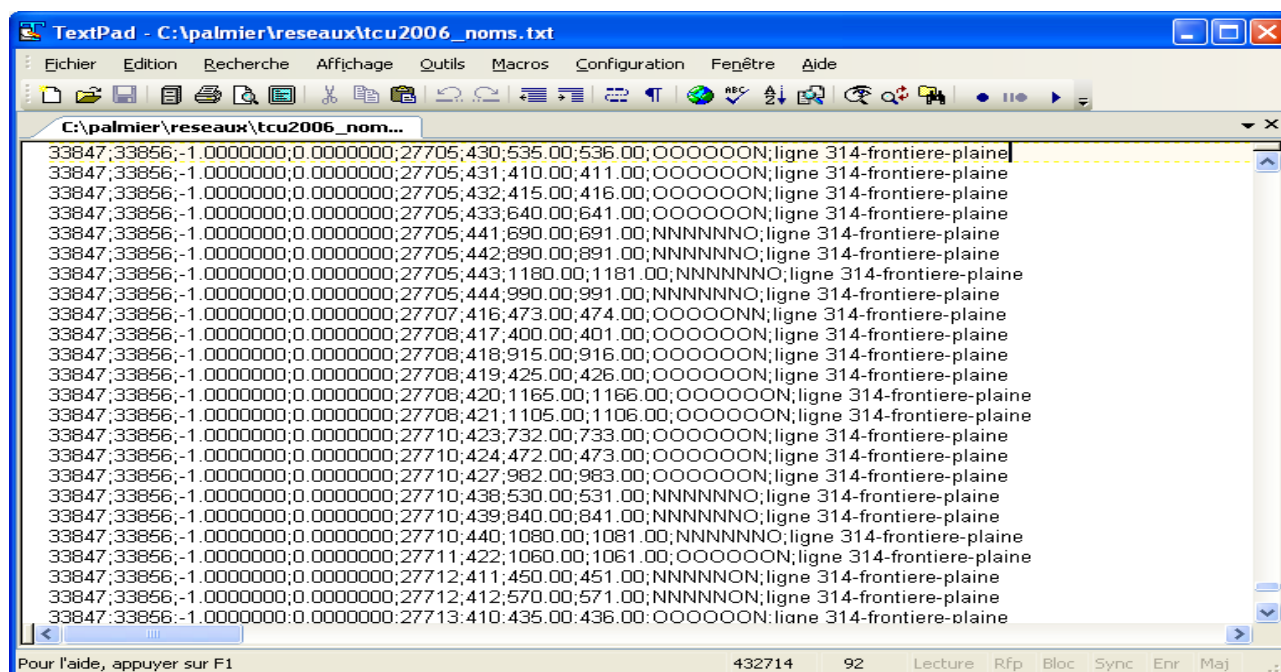
Celle-ci est également valable pour un réseau multimodal à horaires, c'est-à-dire un réseau combinant des services TC avec des tronçons empruntables à pied ou en VP, pouvant dépendre eux aussi d'horaires et de calendrier.

Ce calcul fonctionne également s'il n'y a pas de tronçons à horaires, c'est-à-dire pour calculer une accessibilité à pied, à vélo ou en voiture.

**Depuis la version 1.2.0, MUSLIW permet d'introduire des pénalités et interdictions de mouvements tournants ainsi que de correspondance. Les versions 1.3.0 et suivantes permettent de surcroît la prise en compte de tronçons de type transport individuel dont les temps de parcours dépendent des horaires et du calendrier.**

### 6.1 - Le fichier «Réseau»

Pour définir un réseau TC à horaires dans MUSLIW, il faut disposer d'un fichier du type suivant :



Le fichier réseau est de type «Texte Délimité» avec « ; » comme séparateur.

**NB : Depuis la version 1.4.0.159** MUSLIW gère automatiquement les problèmes de séparateur de décimales '.' ou ','. Il les convertit automatiquement selon les paramètres définis par votre système d'exploitation.

**Pour les versions antérieures à la 1.4.0.159, attention au séparateur décimal !** MUSLIW s'attend à lire, dans les fichiers réseau et matrice, des données numériques définies avec le séparateur de décimales de votre système, généralement « . » ou « , ». Si les données du fichier ont « . » comme séparateur de décimales et que celui de vos paramètres de configuration est « , », ou inversement, le programme va s'interrompre brutalement. Il convient alors ou de changer de type de séparateur dans vos paramètres internationaux, ou bien d'effectuer cette transformation dans le fichier texte directement. Il est recommandé à l'utilisateur d'être particulièrement vigilant lorsque celui-ci souhaite utiliser des réseaux ou des matrices élaborées par d'autres.

Les différents champs dans l'ordre sont :

- numéro de nœud origine du tronçon ; peut-être également une chaîne alphanumérique ;
- numéro de nœud destination du tronçon ; peut-être également une chaîne alphanumérique ;
- temps de parcours du tronçon :
  - nombre de minutes, si le mode de transport sur le tronçon est individuel : VP, marche à pied, vélo ;
  - 1, si le mode de transport sur le tronçon est un transport collectif à horaires.
- longueur du tronçon :
  - longueur ;
  - «0» si la longueur réelle n'est pas disponible.
- numéro de ligne TC :
  - si positif (> 0) :



–identifiant de numéro de ligne (entier), pour les tronçons de transport collectif à horaires ;

•si négatif (< 0) :

–identifiant de période durant laquelle le temps de parcours sera identique. Le tronçon est alors considéré comme étant de type transport individuel variable dans le temps. Cela nécessite donc de décrire la période horaire et le calendrier durant lesquels le temps s'applique. L'identifiant doit être différent pour chacune des périodes comportant un temps de parcours autre. Un exemple de codage peut être trouvé ci-après :

◆-1 pour la période de pointe du matin ;

◆-2 pour la période de pointe du soir ;

◆-3 pour la période creuse de jour ;

◆-4 pour la nuit ;

◆-5 pour les dimanches et jours de fêtes.

•numéro de service :

–identifiant(> 0) de numéro de service (entier), pour les tronçons de transport collectif à horaires ;

–identifiant (> 0) du numéro de plage (entier) pour les tronçons de transport individuel avec prise en compte des horaires et du calendrier ;

–1, pour les tronçons de transport individuel (VP, marche à pied, vélo) sans prise en compte des horaires ou du calendrier. Le tronçon sera accessible 24h/24, 7j/7, toute l'année et présentera invariablement le même temps de parcours.

•heure de départ du nœud origine :

–heure de départ, en minutes (passées minuit), pour les tronçons de transport collectif à horaires (ex: 6h00= 6\*60=360minutes) ;

–heure de début de la période utilisable, en minutes, pour les tronçons de transport individuel avec prise en compte des horaires et du calendrier ;

–1, si le mode de transport est individuel sans prise en compte des horaires et du calendrier : VP, marche à pied, vélo.

•heure d'arrivée au nœud destination :

–heure d'arrivée, en minutes (passées minuit), pour les tronçons de transport collectif à horaires (ex: 6h00= 6\*60=360minutes) ;

–heure de fin de la période utilisable, en minutes, pour les tronçons de transport individuel avec prise en compte des horaires et du calendrier ;

–1, si le mode de transport est individuel : VP, marche à pied, vélo.

•calendrier de circulation du service :

–chaîne de « n » caractères : n= longueur en jours de la période. La circulation du service pour chaque jour de la période est déterminée par « O » pour «circule» ou « N » pour «ne circule pas». Par exemple, pour un service qui ne circule que le 10<sup>ème</sup> jour de la période, la chaîne de caractères correspondante sera composée de 9 « N » puis d'un « O » en 10<sup>ème</sup> position. La

chaîne d'un service qui circule tous les jours sera composée de n « O ». Cette chaîne détermine de la même manière, les jours de circulation pour les tronçons de type transport individuel avec prise en compte des horaires et du calendrier ;

—1, si le mode de transport est individuel : VP, marche à pied, vélo.

•libellé :

—le libellé est utile dans les fichiers résultats pour identifier les tronçons et les itinéraires par leur nom plutôt que par leurs seuls numéros de nœud, de ligne et de service. La partie du libellé avant le premier séparateur vertical « | » détermine généralement l'identifiant de ligne. L'utilisateur n'est pas limité en nombre de caractères.

•type de réseau :

—disponible depuis la version 1.4 le type de réseau permet de définir des paramètres de pondération des temps et coûts différemment selon son type considéré. Ainsi, on peut définir un temps de correspondance de 5 minutes par défaut pour le type 0 de réseau et de 35 minutes pour les tronçons de type Eurostar ou aérien afin de tenir compte de l'enregistrement.

Péage:

—depuis la version 1.5.1, Musliw peut prendre en compte le péage. La valeur est à entrer est celle du coût monétaire sur le tronçon. Musliw calculera ensuite le chemin optimal en tenant compte du péage qui est introduit dans le temps généralisé grâce à un nouveau paramètre de pondération qui peut dépendre du type de tronçon. En sortie Musliw fournira également un élément « toll » qui indique le péage cumulé sur l'origine-destination.

**Attention**, tous les services et tronçons qui ont le même nœud origine et le même nœud destination doivent être listés de manière consécutive dans le fichier réseau, qu'ils soient de type transport individuel ou de type transport collectif.

La limite du nombre de tronçons et de services n'est pas définie. C'est la mémoire de l'ordinateur qui fixera la limite de la taille admissible.

Pour les systèmes d'exploitation 32bits, la taille maximale d'un objet est de 2 Go. Ainsi, MUSLIW ne pourra pas gérer de réseaux plus importants, même si la mémoire vive est supérieure (4 Go est la taille maximale gérable).

A l'inverse, cette limite est beaucoup plus haute pour les systèmes 64bits tels que Windows 7, Linux 64, etc.

La structure interne des données a été modifiée dans la version 1.5 de MUSLIW. Cela permet, à réseau équivalent, d'utiliser bien moins de mémoire et donc de travailler sur des réseaux considérablement plus importants en taille.

L'intérêt majeur de disposer d'un réseau intégrant le calendrier de circulation des services est de pouvoir faire varier les périodes de l'étude d'accessibilité sans avoir à recoder le réseau en conséquence (notamment vis-à-vis des samedis, dimanches, jours de fêtes, etc.).

L'utilisation d'un dictionnaire pour la numérotation des nœuds permet sans dégrader la performance de stocker les numéros de nœuds sous forme de chaînes de caractères. Cela autorise donc les numéros très importants et même l'utilisation des noms de nœuds à la place de numéros.

L'exemple ci-dessous présente la codification des numéros de nœud par des chaînes de caractères et la codification du type de réseau (0 pour tous les tronçons sauf « gare;bruxelles\_E\* » qui est de type 1).

```
depart;gare;5;-1;-1;-1;-1;-1;-1;MARCHE;0  
gare;bruxelles_tgv;-1;-1;1;1;480;530;0000000;TGV|LILLE-BRUX_TGV;0  
gare;bruxelles_E*;-1;-1;2;2;500;532;0000000;E*|LILLE-BRUX_E*;1  
bruxelles_tgv;bruxelles_centre;5;-1;-1;-1;-1;-1;-1;MARCHE;0  
bruxelles_E*;bruxelles_centre;5;-1;-1;-1;-1;-1;-1;MARCHE;0
```

### 6.1.1 -Utilisations du type de réseau

Le type de réseau a deux utilisations essentielles :

- différencier des paramètres de pondération des temps et coûts selon son type ;
- effectuer les calculs de plus courts chemins sur une partie du réseau ;
- pour filtrer les tronçons individuels d'un type particulier il faut rentrer un « cmap » correspondant négatif ;
- pour filtrer les tronçons à horaires d'un type particulier il faut rentrer un « cveh » correspondant négatif.
- permettre des exploitations statistiques par type de réseau

### 6.1.2 -Exemple de codification des tronçons de type individuel tenant compte des horaires et du calendrier

L'extrait ci-dessous fournit un exemple de définition de tronçon dont le temps de parcours dépend de l'horaire et du calendrier :

```
11;151;5.3;0;-1; 1;420;540;OOOOOON
11;151;4.1;0;-3; 1;540;960;OOOOOON
11;151;4.1;0;-3; 2;1140;1200;OOOOOON
11;151;4.6;0;-2; 1;960;1140;OOOOOON
11;151;3.5;0;-4; 1;1200;1440;OOOOOON
11;151;3.5;0;-4; 2;0;420;OOOOOON
11;151;3.6;0;-5; 1;360;1140;NNNNNNO
11;151;3.1;0;-6; 2;0;360;NNNNNNO
11;151;3.1;0;-6; 3;1140;1440;NNNNNNO
```

Les temps sont de :

- 3.5 minutes la nuit (20h-24h ;0h-6h) ;
- 4.1 minutes la journée (9h-16h ;19h-20h) ;
- 4.6 minutes le soir (16h-19h) ;
- 5.3 minutes le matin (7h -9h) et ce du lundi au samedi ;
- 3.1 minutes le dimanche la nuit (0h-6h,20h-24h) ;
- 3.6 minutes le dimanche en journée (6h-20h).

**NB : Ne pas oublier que si l'on rentre les horaires et le calendrier pour des tronçons de type individuel et qu'il existe des périodes de la journée ou des jours dans le calendrier qui ne sont pas définis, le tronçon sera pour ces périodes inaccessible (ce qui se produirait dans le cas d'une route ou rue fermée). Il faut donc veiller à bien couvrir le spectre horaire et calendaire dans la description.**

## 6.2 - Le fichier pénalités et correspondances

Ce fichier regroupe toutes les informations nécessaires pour l'introduction de pénalités et interdictions de mouvements tournants et de correspondance.

Le fichier pénalités et correspondances est de type «Texte Délimité» avec « ; » comme délimiteur.

```
35046;35482;3802;35047;302;1.5  
28325;28264;-1;28347;-1;-1  
28347;28325;-1;28442;-1;0.25
```

Les différents champs dans l'ordre sont dans l'ordre :

- $n_j$  : numéro de nœud du carrefour ou de l'arrêt considéré ;
- $n_i$  : numéro de nœud origine ;
- numéro de ligne du tronçon entrant ( $n_i \rightarrow n_j$ ) ;
- $n_k$  : numéro de nœud final ;
- numéro de ligne du tronçon sortant ( $n_j \rightarrow n_k$ ) ;
- temps de pénalité :
  - 0 : aucune pénalité ;
  - 1 : mouvement ou correspondance interdite ;
  - sinon : valeur du temps de pénalité en minutes.

Si le temps de pénalité est relatif à une correspondance, c'est-à-dire qu'un des deux numéros de ligne n'est pas «-1» le temps de pénalité sera utilisé comme temps de correspondance à la place du temps de correspondance par défaut paramétré lors du calcul et qui sera utilisé par défaut pour toutes les autres correspondances du réseau.

Le fait d'avoir des numéros de ligne distincts pour des périodes différentes décrivant le temps de parcours des tronçons de type transport individuel permet de décrire des pénalités de mouvements tournant ou de correspondance différentes selon les périodes.

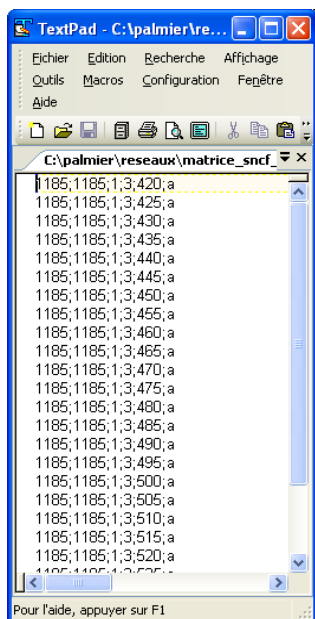
## 6.3 - Le fichier «Matrice»

### 6.3.1 -Format standard

Le fichier matrice est de type «Texte Délimité» avec « ; » comme séparateur.

NB : **Depuis la version 1.4.0.159** MUSLIW gère automatiquement les problèmes de séparateur de décimales '.' ou ','. Il les convertit automatiquement selon les paramètres définis par votre système d'exploitation.

**Pour les versions antérieures à la 1.4.0.159, attention au séparateur décimal !** MUSLIW s'attend à lire, dans les fichiers réseau et matrice, des données numériques définies avec le séparateur de décimales de votre système, généralement « . » ou « , ». Si les données du fichier ont « . » comme séparateur de décimales et que celui de vos paramètres de configuration est « , », ou inversement, le programme va s'interrompre brutalement. Il convient alors ou de changer de type de séparateur dans vos paramètres internationaux, ou bien d'effectuer cette transformation dans le fichier texte directement. Il est recommandé à l'utilisateur d'être particulièrement vigilant lorsque celui-ci souhaite utiliser des réseaux ou des matrices élaborées par d'autres,



Les différents champs dans l'ordre sont :

- nœud origine : numéro du nœud origine : cela peut être n'importe quel nœud du réseau ;
- nœud destination : numéro du nœud destination : cela peut être n'importe quel nœud du réseau ;
- demande : volume de la demande que l'on souhaite affecter entre le nœud origine et le nœud destination ;
- jour : indique le jour de départ ou d'arrivée souhaité. Ce nombre se réfère à la période définie pour chaque service dans le fichier «Réseau». **Le premier jour de la période est le jour 0.** Ainsi, le jour 3 correspond au 4<sup>ème</sup> jour de la période. **Attention, le numéro du jour doit toujours être strictement plus petit que le nombre de jours de la période ;**
- heure : indique l'heure de départ ou d'arrivée souhaitée en minutes. L'heure dans une journée varie entre 0 et 1439 minutes. Le logiciel tolère des heures négatives ou supérieures à 1439. Il incrémentera ou décrémentera pour le calcul du nombre de jours\*1440 pour que l'heure soit comprise entre 0 et 1439 ;
- type d'heure souhaitée :
  - « d » pour départ : le calcul se fera donc du nœud origine en partant au jour et à l'heure indiqués en recherchant de manière itérative les différents successeurs vers le nœud destination ;
  - « a » pour arrivée : le calcul se fera donc du nœud destination pour arriver au nœud destination au jour et à l'heure indiqués en recherchant de manière itérative les différents prédécesseurs vers le nœud origine.

Il n'y a pas de limite de taille pour le nombre de lignes du fichier matrice.

Lors de la recherche du plus court chemin, un tronçon est pris en compte dans l'itinéraire s'il permet de se rendre à un tronçon successeur avec un coût strictement inférieur à celui du chemin retenu comme le plus court à l'itération en cours. Ainsi, si plusieurs itinéraires sont équivalents le premier trouvé sera celui retenu ; toute la demande de la ligne du fichier matrice correspondante lui sera affectée.

### 6.3.2 -Format avancé

MUSLIW offre la possibilité de détailler pour chaque ligne du fichier matrice les paramètres du calcul à prendre (en bleu) et même éventuellement les paramètres de l'algorithme et des sorties (en vert).

Cependant, si les paramètres de calcul (partie bleue) peuvent être indiqués sans que ne le soient ceux de l'algorithme et des sorties (partie verte), la réciproque n'est pas vraie. Afin de pouvoir indiquer les paramètres d'algorithme et de sortie en mode batch, il est également nécessaire d'indiquer les paramètres de calcul.

Ainsi que le montre l'exemple ci-dessous, la finesse du paramétrage peut varier selon les lignes du fichier matrice. En revanche, pour être actif, chaque bloc de couleur doit être complet.

```
1546;1368;1;1;420;a; 1;1.5;3;5;1;2;120,0;30,2
1546;1368;1;1;435;a; 1;1.5;3;5;1;2;120,0;30,2, true;0;0;50;10000;2
1546;1368;1;1;450;a; 1;1.5;3;5;1;2;120,0;30,2, true;0;0;50;10000;2
1546;1368;1;1;465;a
1546;1368;1;1;480;a
1546;1368;1;1;495;a; 1;1.5;3;5;1;2;120,0;30,2, true;1;0;50;10000;2
1546;1368;1;1;510;a; 1;1.5;3;5;1;2;120,0;30,2
1546;1368;1;1;525;a; 1;1.5;3;5;1;2;120,0;30,2, true;0;0;50;10000;2
1546;1368;1;1;540;a; 1;1.5;3;5;1;2;120,0;30,2, true;0;0;50;10000;2
```

Bloc bleu dans l'ordre :	Bloc vert dans l'ordre :
•Poids TC ;	•Sortie chemins (true/false) ;
•Poids ATT ;	•Temps détaillés (0 Aucun tronçon, 1 Sans tronçon TC, 2 avec tronçons TC) ;
•Poids MAP ;	•Algorithme (0 GGA à intervalles, 1 Dijkstra à intervalles) ;
•Poids COR ;	•Paramètre de l'algorithme ;
•Coef TMAP ;	•Nombre d'intervalles ;
•Temps COR mini ;	•Paramètre puissance.
•Temps COR maxi ;	
•Nb jours.	
•Temps MAP maximal	
•Poids péage	

Comme les paramètres de pondération de calcul peuvent être différenciés selon le type de réseau, ils peuvent être rentrés séparés par un « | ». Ex 1,1.5;3;2;1;2|5|35;0 spécifie un temps de correspondance différent par type de réseau (2 minutes pour le type « 0 », 5 minutes pour le type « 1 », et 35 minutes pour le type « 2 »).

## 6.4 - Optimiser les temps de calcul

Pour minimiser les temps de calcul, il convient de trier le fichier matrice selon les règles ci-dessous :

- pour des calculs à partir d'un **point de départ**, le fichier doit être trié par :

- Origine ;

- Jour ;

- Heure.

- pour des calculs à partir d'un **point d'arrivée**, le fichier doit être trié par :

- Destination ;

- Jour ;

- Heure.

En effet, MUSLIW calcule les plus courts chemins d'un point à l'ensemble des autres points, donc si l'origine, le jour et l'heure de départ ou d'arrivée souhaitée sont identiques entre deux lignes consécutives du fichier matrice, il n'est pas nécessaire de recalculer l'ensemble des plus courts chemins, d'où des gains en temps de calcul importants pour les grands réseaux et les fichiers matrice de taille importante.

## 6.5 - La procédure de calcul

### 6.5.1 -Cas courant

La procédure se lance via la barre de menu « Procédures > Affectation TC à horaires ». Il faut alors renseigner la boîte de dialogue qui apparaît, comme représentée ci-dessous.



Le paramétrage de cette boîte se fait comme suit :

- Réseau : permet de rechercher et charger le fichier « réseau » qui sera utilisé pour le calcul ;
- Pénalités (optionnel) : permet de rechercher et charger le fichier de pénalités et correspondances qui sera utilisé pour le calcul ;
- Matrice : permet de rechercher et charger le fichier « matrice » qui sera utilisé pour le calcul ;
- Poids TC : coefficient de pondération du temps TC à bord du véhicule ;
- Poids ATT : coefficient de pondération des temps d'attente TC ;
- Poids MAP : coefficient de pondération des temps en transport individuel (Marche à Pied (MAP), VP, Vélo) ;
- Max: Temps individuel cumulé maximal
- Poids COR : coefficient de pondération des temps de correspondance TC ;
- Coef TMAP : coefficient multiplicateur uniforme de tous les temps en transport individuel. Effectué une fois avant le lancement de tous les calculs. Dans le cas où les temps de transport individuel ont été définis à partir d'une vitesse constante, ce coefficient permet de tester une vitesse de marche à pied différente sans avoir à modifier les temps de parcours dans la description du réseau. Par exemple, pour un réseau dont les temps de marche à pied ont été définis avec une vitesse de 4km/h, si l'on veut effectuer des calculs pour des personnes marchant à 5km/h, on prendra un coefficient TMAP égal à 1,25 sans avoir à changer les temps de parcours dans la description du réseau ;
- Temps COR Mini : temps de correspondance minimal. C'est le temps minimum possible entre l'arrivée à un arrêt et la possibilité de monter dans un véhicule. Ce temps sera pondéré par « Poids COR » et pris en compte dans le temps généralisé ;
- Temps COR Maxi : délai de correspondance maximal. C'est le temps maximum admissible entre l'arrivée à un arrêt et la possibilité de monter dans le véhicule suivant. Au-delà de cette valeur la correspondance est jugée non attractive et ne sera pas prise en compte dans la recherche de plus court chemin ;

- Nb jours : détermine le nombre de jours durant lesquels l'algorithme va rechercher les plus courts chemins. « 0 » indique que seuls les services circulant le jour indiqué dans l'origine ou la destination concernée seront examinés. La recherche des itinéraires à cheval sur plusieurs jours ou très peu fréquents comme les lignes d'autocars régulières internationales ou encore relatif à des questionnements de type « accessibilité en restant N jours à destination » ne peut pas être effectuée ;
- Péage: pondération du péage. Cette pondération peut être différente selon le type de tronçon.
- Algorithme : *Graph Growth Algorithm* à intervalles ; Dijkstra à intervalles ;
- Paramètre : 200 (voir paramètres de l'algorithme) ;
- Max classes : 10000 (voir paramètres de l'algorithme) ;
- Puissance : 2 (voir paramètres de l'algorithme) ;
- Sortie chemins : en activant cette case, MUSLIW écrira, pour chaque origine-destination, l'itinéraire, c'est-à-dire la succession de tronçons et de services avec les différentes composantes de temps intermédiaires reconstituant l'itinéraire. En cas de nombreuses lignes dans le fichier matrice, ce fichier peut rapidement être de taille très importante ;
- Temps détaillés : en cas de nombreuses lignes dans le fichier matrice, le fichier de sortie avec les temps détaillés peut rapidement être de taille très importante. Il est donc possible de choisir les options suivantes :
  - aucun tronçon : aucune sortie ;
  - sans tronçon TC : écriture pour chaque origine-destination des temps de tous les tronçons de type transport individuel accessibles, mais aucun tronçon à horaires ;
  - avec tronçon TC : écriture pour chaque origine-destination des temps de tous les tronçons accessibles de type transport individuel ou à horaires.
- Demi-tours interdits : l'activation de cette commande interdit par défaut la possibilité d'effectuer des demi-tours (même nœud origine et même nœud final dans un mouvement tournant ou une correspondance). Ces interdictions seront aussi effectives pour les tronçons de transport à horaires ;
- Sortie trafics par service : cette option permet de générer un fichier résultats fournissant le détail des volumes de flux par service, pour ceux non nuls ;
- Sortie virage et correspondances : cette option permet de générer un fichier résultat précisant pour chaque carrefour ou nœud de correspondance, les mouvements tournants et/ou les correspondances ligne à ligne.

En cliquant sur « OK », MUSLIW ouvre un boîte de dialogue « Enregistrer sous ». L'utilisateur devra alors sélectionner un répertoire et indiquer un nom de fichier qui sera utilisé pour l'écriture des fichiers résultats.

Il est conseillé de choisir un nom de fichier sans extension car MUSLIW rajoutera au nom choisi le type de résultat et l'extension « TXT ».

MUSLIW lit et écrit les derniers paramètres par défaut utilisés pour un calcul et les stocke dans un fichier « ,INI ».

Ainsi, lors du redémarrage de l'outil, les précédents paramètres seront remplis par défaut.

## 6.5.2 -Paramétrage avancé

Le paramétrage avancé permet d'appliquer des pondérations et des valeurs qui sont fonction du type de réseau.

Les paramètres pouvant être définis selon le type du réseau sont :

- Poids TC ;
- Poids ATT ;
- Poids MAP ;
- Poids COR
- Coef TMAP ;
- Temps COR;
- Péage

Le paramétrage s'effectue en séparant les différentes valeurs par un « | ». Ainsi, dans l'exemple ci-dessus, « 5|35 » signifie que le temps de correspondance est de 5 minutes si le type de réseau est « 0 » (1ère valeur) et de 35 minutes si le type de réseau est « 1 ». Lorsqu'il n'y a pas de valeur particulière définie pour un type en question, MUSLIW prend la valeur par défaut, c'est à dire la 1ère valeur qui correspond à celui de type 0.

Si aucun type n'est rentré le type de tous les tronçons est défini par défaut à « 0 ».

## 6.5.3 -Les fichiers « Résultats »

L'outil MUSLIW restitue en sortie quatre fichiers de résultats qui fournissent des informations différentes :

- temps et volume par origine-destination, pour celles demandées dans la matrice en entrée ;

- tous les tronçons accessibles dans une limite de temps fixée par origine ou destination, pour toutes les origines et destinations de la matrice ;
- les volumes par tronçon et segment de ligne lorsque ces volumes sont non nuls ;
- les chemins pour les origine-destination demandées dans la matrice en entrée.

## 6.6 - <NOM\_FICHIER>\_OD.TXT

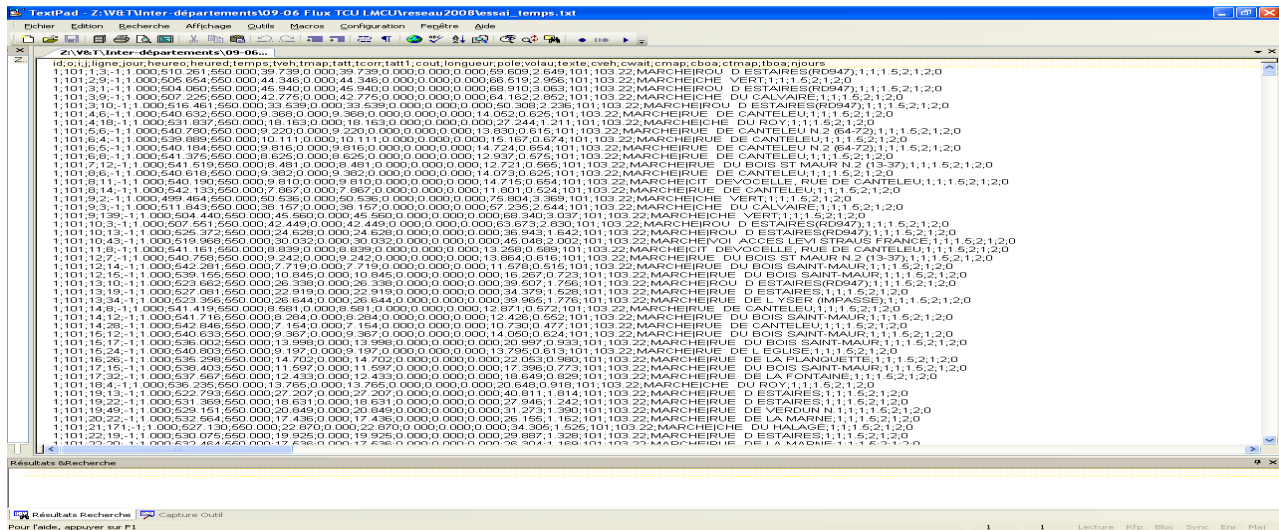
```
id;o;d;jour;heureo;heured;temps;tveh;tmap;tatt;tcorr;tatt1;cout;longueur;pole;volau;cveh;cwait;cmapi;cboai;ctmap;tboai;njours;texte
1;13657;101;1.000;394.348;550.000;155.652;53.000;11.691;90.961;6.000;81.961;173.498;0.779;37802;103.22;1;1;1.5;2;1;2;0;MARCHE|ligne Citadine 1|ligne 54|ligne 63|MARCHE
2;2495;129;0.000;406.546;450.000;43.454;26.000;14.552;2.903;2.000;0.903;54.730;0.970;36881;100.58;1;1;1.5;2;1;2;0;MARCHE|ligne 319|MARCHE
3;1019;129;0.000;421.545;485.000;63.455;12.000;13.553;37.903;2.000;35.903;74.232;0.904;36828;75.41;1;1;1.5;2;1;2;0;MARCHE|ligne 319|MARCHE
4;183;129;0.000;650.998;665.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
5;1349;129;0.000;966.992;1085.000;118.008;59.000;6.105;52.903;2.000;50.903;125.061;0.407;36564;35.11;1;1;1.5;2;1;2;0;MARCHE|ligne 214|MARCHE
6;12660;129;0.000;1071.737;1170.000;98.263;62.000;9.361;26.903;6.000;19.903;114.943;0.624;35013;121.00;1;1;1.5;2;1;2;0;MARCHE|ligne 1|ligne 2|ligne 64|MARCHE
7;183;129;1.000;405.998;420.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
8;477;129;1.000;484.278;530.000;45.722;0.000;45.722;0.000;0.000;0.000;68.583;3.048;129;75.41;1;1;1.5;2;1;2;0;MARCHE
9;477;129;1.000;540.354;595.000;54.646;10.000;11.744;32.903;2.000;30.903;64.518;0.783;36772;63.59;1;1;1.5;2;1;2;0;MARCHE|ligne 63|MARCHE
10;183;129;1.000;590.998;605.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
11;477;129;1.000;559.278;605.000;45.722;0.000;45.722;0.000;0.000;0.000;68.583;3.048;129;35.11;1;1;1.5;2;1;2;0;MARCHE
12;8029;129;1.000;493.394;635.000;141.606;62.000;7.703;71.903;6.000;62.903;157.457;0.514;35978;142.66;1;1;1.5;2;1;2;0;MARCHE|ligne 73|ligne 2|ligne 64|MARCHE
13;1019;129;1.000;571.846;660.000;88.154;20.000;20.722;47.432;4.000;10.903;106.515;1.381;36701;96.80;1;1;1.5;2;1;2;0;MARCHE|ligne 63 R|MARCHE|ligne 64 R|MARCHE
14;1019;129;1.000;571.846;662.000;90.154;20.000;20.722;49.432;4.000;12.903;108.515;1.381;36701;96.80;1;1;1.5;2;1;2;0;MARCHE|ligne 63 R|MARCHE|ligne 64 R|MARCHE
15;183;129;1.000;660.998;675.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
16;4067;129;1.000;684.881;775.000;90.120;36.000;18.217;35.903;2.000;33.903;103.228;1.214;36752;100.55;1;1;1.5;2;1;2;0;MARCHE|ligne 64|MARCHE
17;1019;129;1.000;705.846;785.000;79.154;14.000;8.251;56.903;2.000;54.903;87.279;0.550;36701;63.59;1;1;1.5;2;1;2;0;MARCHE|ligne 63 R|MARCHE
18;477;129;1.000;849.278;895.000;45.722;0.000;45.722;0.000;0.000;0.000;68.583;3.048;129;63.59;1;1;1.5;2;1;2;0;MARCHE
19;183;129;1.000;915.998;930.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
20;183;129;1.000;960.998;975.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;188.00;1;1;1.5;2;1;2;0;MARCHE
21;183;129;1.000;985.998;1000.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
22;9748;129;1.000;813.300;1015.000;201.700;73.000;7.797;120.903;6.000;113.903;217.598;0.520;35027;75.41;1;1;1.5;2;1;2;0;MARCHE|ligne 2|ligne 1|ligne 63|MARCHE
23;183;129;1.000;1030.998;1045.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
24;4067;129;1.000;1095.881;1165.000;69.119;34.000;18.217;16.903;2.000;14.903;82.228;1.214;36752;100.55;1;1;1.5;2;1;2;0;MARCHE|ligne 64|MARCHE
25;183;129;1.000;1220.998;1235.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
26;183;129;1.000;1220.998;1235.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
27;183;129;1.000;1250.998;1265.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;188.00;1;1;1.5;2;1;2;0;MARCHE
28;183;129;2.000;340.998;355.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
29;1019;129;2.000;421.545;485.000;63.455;12.000;13.553;37.903;2.000;35.903;74.232;0.904;36828;188.00;1;1;1.5;2;1;2;0;MARCHE|ligne 319|MARCHE
30;477;129;2.000;476.354;505.000;28.646;5.000;18.994;4.653;2.000;2.653;42.143;1.266;36772;63.59;1;1;1.5;2;1;2;0;MARCHE|ligne 328|MARCHE
31;183;129;2.000;500.998;515.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;100.55;1;1;1.5;2;1;2;0;MARCHE
32;477;129;2.000;492.278;538.000;45.722;0.000;45.722;0.000;0.000;0.000;68.583;3.048;129;63.59;1;1;1.5;2;1;2;0;MARCHE
33;477;129;2.000;773.354;805.000;31.646;3.000;18.994;9.653;2.000;7.653;45.143;1.266;36772;63.59;1;1;1.5;2;1;2;0;MARCHE|ligne 214|MARCHE
34;183;129;2.000;805.998;820.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
35;183;129;2.000;925.998;940.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
36;183;129;2.000;1015.998;1030.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
37;183;129;2.000;1060.998;1075.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
38;183;129;2.000;1195.998;1210.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;35.11;1;1;1.5;2;1;2;0;MARCHE
39;477;129;3.000;459.354;490.000;30.646;5.000;18.994;6.653;2.000;4.653;44.143;1.266;36772;121.00;1;1;1.5;2;1;2;0;MARCHE|ligne 328|MARCHE
40;360;129;3.000;448.681;495.000;46.319;7.000;17.445;21.874;2.000;19.874;59.041;1.163;36660;188.00;1;1;1.5;2;1;2;0;MARCHE|ligne 214|MARCHE
41;686;129;3.000;448.639;505.000;56.161;22.000;20.287;13.874;4.000;9.874;74.305;1.352;36721;164.77;1;1;1.5;2;1;2;0;MARCHE|ligne 68|ligne 328|MARCHE
42;183;129;3.000;685.998;700.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;63.59;1;1;1.5;2;1;2;0;MARCHE
43;183;129;3.000;690.998;705.000;14.002;0.000;14.002;0.000;0.000;0.000;21.003;0.933;129;141.00;1;1;1.5;2;1;2;0;MARCHE
44;6204;129;3.000;640.131;785.000;144.869;62.000;25.966;56.903;2.000;54.903;161.852;1.731;35840;121.00;1;1;1.5;2;1;2;0;MARCHE|ligne 63 R|MARCHE
```

Le fichier des résultats par OD se situe dans le répertoire que l'utilisateur a spécifié en cliquant sur « OK ». C'est un fichier de type « Texte délimité » avec « ; » comme séparateur. Il contient une ligne par enregistrement du fichier « matrice » avec comme champs :

- id : numéro séquentiel correspondant au numéro de ligne du fichier matrice ;
- o : numéro de nœud origine du déplacement ;
- d : numéro de nœud destination du déplacement ;
- jour : jour du déplacement ;
- heureo : heure de départ du déplacement :
  - si type d'heure = « d », c'est l'heure de départ souhaitée indiquée dans le fichier matrice ;
  - si type d'heure = « a », c'est l'heure de départ au plus tard pour arriver à destination avant l'heure d'arrivée ;

- **heured** : heure d'arrivée : heure d'arrivée du déplacement :
  - si type d'heure = « d », c'est l'heure d'arrivée au plus tôt pour arriver à destination en partant à l'heure de départ ;
  - si type d'heure = « a », c'est l'heure d'arrivée souhaitée indiquée dans le fichier matrice ;
- **temps** : temps de parcours entre le nœud origine et le temps destination (minutes) :  
 $\text{temps} = [\text{heure arrivée}] - [\text{heure départ}]$  ;
- **tveh** : temps passé à bord des transports collectifs (minutes) ;
- **tmap** : temps de marche à pied ou de transport individuel (minutes) ;
- **tatt** : temps d'attente à un arrêt avant de monter dans un transport en commun (minutes) ;
- **tcorr** : temps passé dans les correspondances de TC, ce temps est égal au nombre de trajets réalisés en TC multiplié par le temps de correspondances ;
- **ncorr** : nombre de correspondances. Ce nombre représente le nombre de montées dans un véhicule de transport en commun dont le service est décrit par des horaires ;
- **tatt1** : temps d'attente avant de monter dans le premier TC (sens d) ou temps d'attente entre l'heure d'arrivée effective et l'heure d'arrivée souhaitée (sens a) ;
- **coût** : temps généralisé minimisé par l'algorithme de recherche de plus courts chemins.  
 $[\text{temps généralisé}] = [\text{Poids TC}] * [\text{temps véhicule}] + [\text{Poids ATT}] * [\text{temps attente}] + [\text{Poids MAP}] * [\text{temps marche}] + [\text{nombre trajets TC}] * [\text{temps COR}]$  ;
- **longueur** : longueur cumulée ;
- **pôle** : numéro de nœud du premier point d'intermodalité (passage du réseau de transports individuels au réseau à horaires). Si tout l'itinéraire se fait sur le réseau individuel, le pôle a la valeur du nœud origine ou destination selon le sens du calcul ;
- **volau** : volume de demande sur le dernier tronçon de l'itinéraire ;
- **texte** : liste des lignes de transports collectifs utilisées sur l'itinéraire. Pour que cette fonctionnalité fournisse des résultats convenables ou exploitables, les noms des tronçons TC à horaires doivent commencer par un identifiant (qui peut être de type caractères) de la ligne suivi d'un « - ». C'est cet identifiant qui sera repris dans la succession des lignes empruntées, faute de quoi, l'identifiant utilisé sera la chaîne de caractères définie du début du nom du tronçon jusqu'au premier « - ».
- **nbpop** : nombre d'itérations lors du calcul des plus courts chemins. Cet indicateur permet d'optimiser les paramètres de l'algorithme pour un temps de calcul minimal. Pour une origine-destination équivalente, une valeur nbpop plus petite indique un temps de calcul plus optimal ;
- **ttoll** : péage cumulé le long de l'itinéraire.

## 6.7 - <NOM\_FICHIER>\_TEMPS.TXT



Le fichier résultat des temps ne sera renseigné que si la case « Sortie temps » a été activée. La structure de ce fichier est identique à celle du fichier origines-destinations à l'exception du dernier champ sur les lignes empruntées qui est absent.

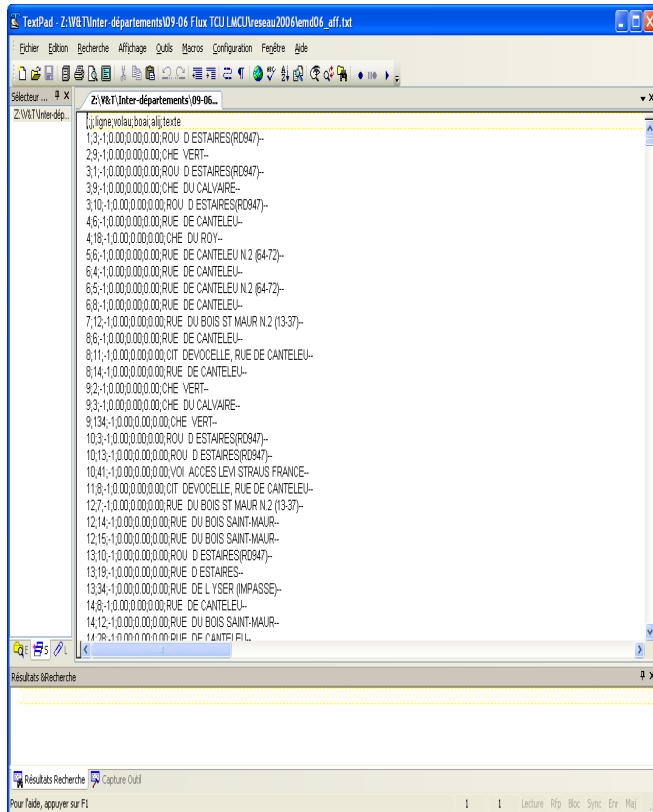
La première ligne indique le nom des champs.

Puis, le fichier donne pour chaque ligne du fichier matrice, l'ensemble des tronçons du réseau qui sont accessibles avec les paramètres de calcul du temps généralisé définis dans la boîte de dialogue décrite au 4.3. Ce fichier peut donc devenir rapidement très volumineux pour les réseaux de grande taille :

- id : numéro séquentiel correspondant au numéro de ligne du fichier matrice ;
- o : numéro de nœud origine (sens d) ou destination (sens a) du déplacement ;
- i : numéro de nœud origine du tronçon accessible ;
- j : numéro de nœud fin du tronçon accessible ;
- numtrc : numéro identifiant du tronçon
- jour : jour du déplacement ;
- heureo : heure de départ du déplacement :
  - si type d'heure = « d », c'est l'heure de départ souhaitée indiquée dans le fichier matrice ;
  - si type d'heure = « a », c'est l'heure de départ au plus tard pour arriver à destination avant l'heure d'arrivée ;
- heured : heure d'arrivée du déplacement :
  - si type d'heure = « d », c'est l'heure d'arrivée au plus tôt pour arriver à destination en partant à l'heure de départ ;
  - si type d'heure = « a », c'est l'heure d'arrivée souhaitée indiquée dans le fichier matrice ;

- temps : temps de parcours entre le nœud origine et le temps destination (minutes). Il est donné par la formule  $\text{temps} = [\text{heured}] - [\text{heureo}]$  ;
- tveh : temps passé à bord des transports collectifs (minutes) ;
- tmap : temps de marche à pied ou de transport individuel (minutes) ;
- tatt : temps d'attente à un arrêt avant de monter dans un transport en commun (minutes) ;
- tcorr : temps passé dans les correspondances de TC, ce temps est égal au nombre de trajets réalisés en TC multiplié par le temps de correspondances ;
- ncorr : nombre de correspondances. Ce nombre correspond au nombre de montées dans un véhicule de transport en commun dont le service est décrit par des horaires ;
- tatt1 : temps d'attente avant de monter dans le premier TC (sens d) ou temps d'attente entre l'heure d'arrivée effective et l'heure d'arrivée souhaitée (sens a) ;
- coût : temps généralisé minimisé par l'algorithme de recherche de plus courts chemins.  
 $[\text{temps généralisé}] = [\text{Poids TC}] * [\text{temps véhicule}] + [\text{Poids ATT}] * [\text{temps attente}] + [\text{Poids MAP}] * [\text{temps marche}] + [\text{nombre trajets TC}] * [\text{temps COR}]$  ;
- longueur : longueur cumulée ;
- pôle : numéro de nœud du premier point d'intermodalité (passage du réseau de transport individuel au réseau à horaires). Si tout l'itinéraire se fait sur le réseau individuel, le pôle a la valeur du nœud origine ou destination selon le sens du calcul ;
- volau : volume de demande sur le dernier tronçon de l'itinéraire ;
- précédent : Numéro du tronçon précédent. Ce champ permet de reconstituer les itinéraires en remontant successivement les tronçons précédents
- type: Indicateur du type de tronçon
- toll: Péage cumulé le long de le l'itinéraire

## 6.8 - <NOM\_FICHER>\_AFF.txt



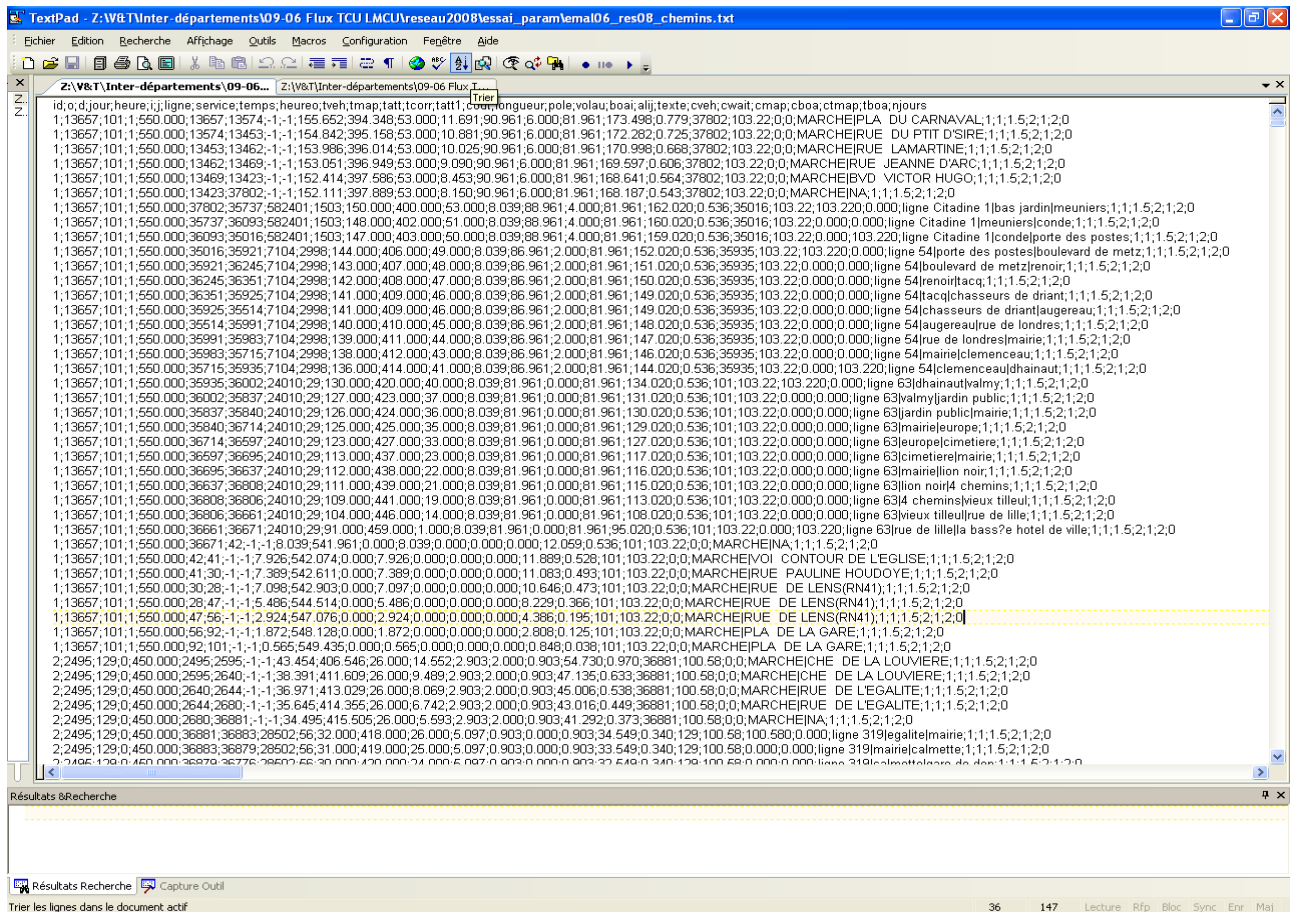
Le fichier des résultats d'affectation est de type « Texte délimité » avec « ; » comme séparateur. Il contient un enregistrement par tronçon et segment de ligne et comporte comme champs :

- nœud origine : numéro de nœud début du tronçon ;
- nœud fin : numéro de nœud fin du tronçon ;
- ligne :
  - identifiant de ligne TC si le mode est à horaires ;
  - 1 si le mode est de type transport individuel ;
- volau : volume affecté sur le tronçon ;
- boai : nombre de montées au nœud initial ;
- alij : nombre de descentes au nœud final ;
- texte : nom du tronçon.
- type: type de tronçon
- toll: péage du tronçon



## 6.9 - <NOM\_FICHIER>\_CHEMINS.TXT

Le fichier résultat des chemins est de type « Texte délimité » avec « ; » comme séparateur. Il décrit de manière détaillée l'itinéraire de chaque enregistrement du fichier « matrice ».



Il contient un enregistrement par tronçon et segment de ligne décrivant l'itinéraire et comporte comme champs :

- id : numéro séquentiel correspondant au numéro de ligne du fichier matrice nœud origine : numéro de nœud origine du déplacement ;
- o : numéro de nœud destination du déplacement ;
- d : numéro de nœud destination du déplacement ;
- jour : jour du déplacement ;
- heure : heure de départ ou d'arrivée souhaitée ;
- i : numéro de nœud début du tronçon ;
- j : numéro de nœud fin du tronçon ;
- ligne : identifiant de ligne TC si le mode est à horaires ;

—1 si le mode est de type transport individuel ;

•service :

—identifiant de service TC si le mode est à horaires ;

—1 si le mode est de type transport individuel ;

•temps : temps de parcours entre le nœud origine et le temps destination (minutes) :  
 $\text{temps} = [\text{heured}] - [\text{heureo}]$  ;

•heureo : heure à l'origine du tronçon si point de départ ou à l'arrivée si point d'arrivée ;

•tveh : temps passé à bord des transports collectifs (minutes) ;

•tmap : temps de marche à pied ou de transport individuel (minutes) ;

•tatt : temps d'attente à un arrêt avant de monter dans un transport en commun (minutes) ;

•tcorr : temps passé dans les correspondances de TC, ce temps est égal au nombre de trajets réalisés en TC multiplié par le temps de correspondances ;

•ncorr : nombre de correspondances. Ce nombre correspond au nombre de montées dans un véhicule de transport en commun dont le service est décrit par des horaires ;

•tatt1 : temps d'attente avant de monter dans le premier TC (sens d) ou temps d'attente entre l'heure d'arrivée effective et l'heure d'arrivée souhaitée (sens a) ;

•coût : temps généralisé minimisé par l'algorithme de recherche de plus courts chemins.  
 $[\text{temps généralisé}] = [\text{Poids TC}] * [\text{temps véhicule}] + [\text{Poids ATT}] * [\text{temps attente}] + [\text{Poids MAP}] * [\text{temps marche}] + [\text{nombre trajets TC}] * [\text{temps COR}]$  ;

•longueur : longueur cumulée ;

•pôle : numéro de nœud du premier point d'intermodalité (passage du réseau de transport individuel au réseau à horaires). Si tout l'itinéraire se fait sur le réseau individuel, le pôle a la valeur du nœud origine ou destination selon le sens du calcul ;

•volau : volume de demande sur le tronçon pour le déplacement considéré ;

•boai : nombre de montées au nœud origine du tronçon pour le déplacement considéré ;

•alij : nombre de descentes au nœud destination du tronçon pour le déplacement considéré ;

•texte : nom du tronçon ;

•type: type de tronçon

•toll: détail du péage cumulé le long du chemin

En consultant ce fichier et pour toutes les lignes ayant les mêmes nœuds origine et destination, il est possible de connaître l'itinéraire emprunté sur l'origine-destination choisie, de connaître les volumes et montées descentes par service et de faire des analyses détaillées (arborescences, chemins empruntés, ...).

## 6.10 - <NOM\_FICHER>\_LOG.TXT

MUSLIW fournit maintenant un petit fichier log qui récapitule des informations pouvant être utiles lorsque l'on souhaite retrouver les fichiers et les paramètres utilisés ou évaluer les différences de temps de calcul par rapport à des paramètres d'algorithme. Un exemple de fichier Log peut être trouvé ci-dessous :

```
Version: 1.5.1.211
Début de la procédure: vendredi 04 mars 2011 11:39:16.968
Paramètres par défaut:
Temps correspondance minimum par défaut:2
Temps correspondance maximum par défaut:120
Pondération correspondance:5
Pondération attente:1
Pondération temps à horaires:1
Pondération temps individuel:1.5
Coefficient temps individuel:1
Temps individuel maximal:30
Pondération du péage:0
Nombre de jours:1
Interdiction demi-tours:False
Algorithme:0
Nombre d'intervalles:10000
Paramètre d'échelle de l'algorithme:15
Paramètre exposant de l'algorithme:2
Sortie chemins:True
Sortie temps:1
Noms fichiers sortie:C:\palmier\reseaux\tutu
Réseau:C:\palmier\reseaux\transpole2006_musliw.txt
Noeuds:32084
Liens:102580
Services horaires:359508
Matrice:C:\palmier\reseaux\matrice_dt_2.txt
Début Calcul: vendredi 04 mars 2011 11:39:23.906
Fin Calcul: vendredi 04 mars 2011 11:39:35.203
Temps Calcul:11.296875 sec
```

Le fichier log reprend :

- la version du Musliw
- les paramètres de calcul par défaut (les paramètres particuliers par origine-destination sont dans les fichiers résultats) ;
- les chemins des fichiers d'entrée et de sortie ;
- les éléments de taille du réseau (noeuds, liens, mouvements tournants et correspondances, services) ;
- des éléments pour évaluer le temps de calcul ;
- les origines et/ou destinations inaccessibles.

## 6.11 - <NOM\_FICHIER>\_SERVICES.TXT

Le fichier résultat des volumes par service est de type « Texte délimité » avec « ; » comme séparateur. Il décrit de manière détaillée l'ensemble des volumes de flux et de montées-descentes par service.

```
i;j;ligne;service;hd;hf;regime;volau;boia;alij;texte;type
908272;688583;1026;1026;1065;1145;OOOOOOO;1.00;0.00;0.00;ZRH DUSSwiss/Crossair;2
8000082;8000085;9510136;9510136;1291;1297;OOOOOOO;1.00;0.00;0.00;Düsseldorf Flughafen Düsseldorf Hbf;1
8000084;8003553;9510136;9510136;1359;1365;OOOOOOO;1.00;0.00;0.00;Düren Langerwehe;1
8000085;8001584;9510136;9510136;1300;1305;OOOOOOO;1.00;0.00;0.00;Düsseldorf Hbf Düsseldorf-Benrath;1
8000178;8000084;9510136;9510136;1348;1358;OOOOOOO;1.00;0.00;0.00;Horrem Düren;1
8000207;8000208;9510136;9510136;1335;1339;OOOOOOO;1.00;0.00;0.00;Köln Hbf Köln-Ehrenfeld;1
8000208;8000178;9510136;9510136;1340;1347;OOOOOOO;1.00;0.00;0.00;Köln-Ehrenfeld Horrem;1
8000209;8003368;9510136;9510136;1322;1327;OOOOOOO;1.00;0.00;0.00;Köln-Mülheim Köln Messe/Deutz;1
8000348;8000406;9510136;9510136;1377;1382;OOOOOOO;1.00;0.00;0.00;Stolberg(Rheinl)Hbf Aachen-Rothe Erde;1
8000406;8000001;9510136;9510136;1383;1387;OOOOOOO;1.00;0.00;0.00;Aachen-Rothe Erde Aachen Hbf;1
8001584;8006713;9510136;9510136;1306;1314;OOOOOOO;1.00;0.00;0.00;Düsseldorf-Benrath Leverkusen Mitte;1
8001886;8000348;9510136;9510136;1373;1376;OOOOOOO;1.00;0.00;0.00;Eschweiler Hbf Stolberg(Rheinl)Hbf;1
8003368;8000207;9510136;9510136;1329;1332;OOOOOOO;1.00;0.00;0.00;Köln Messe/Deutz Köln Hbf;1
8003553;8001886;9510136;9510136;1366;1372;OOOOOOO;1.00;0.00;0.00;Langerwehe Eschweiler Hbf;1
8006713;8000209;9510136;9510136;1315;1321;OOOOOOO;1.00;0.00;0.00;Leverkusen Mitte Köln-Mülheim;1
```

- i : noeud origine ;
- j : noeud destination ;
- ligne : numéro de ligne ;
- service : numéro de service ;
- hd : heure de départ du noeud i ;
- hf : heure d'arrivée au noeud j ;
- regime : calendrier de circulation du service ;
- volau : volume de flux du service ;
- boia : nombre de montées au noeud i ;
- alij : nombre de descentes au noeud j ;
- texte : libellé du tronçon ;
- type : type du tronçon.

Seuls les services dont le volume de flux n'est pas nul sont inscrits dans le fichier.

## 6.12 - <NOM\_FICHER>\_TURNS.TXT

Le fichier résultat des chemins est de type « Texte délimité » avec « ; » comme séparateur. Il décrit de manière détaillée l'ensemble des mouvements tournants et correspondances ligne à ligne à un carrefour ou un arrêt de transport en commun.

```
j;i;lignei;textei;k;lignek;textek;volau;
8000082;8000085;9510115; D sseldorf Hbf D sseldorf Flughafen;688583;-1; DUS - Duesseldorf Flughafen ;1;a
688583;8000082;-1; Duesseldorf Flughafen - DUS ;908272;1026;ZRH DUSSwiss/Crossair;1;a
8000084;8003553;9510141; Langerwehe D ren;8000178;9510136; Horrem D ren;1;a
8000207;8000208;9511313; K ln-Ehrenfeld K ln Hbf;8003368;9510136; K ln Messe/Deutz K ln Hbf;1;a
8000406;8000001;9510139; Aachen Hbf Aachen-Rothe Erde;8000348;9510136; Stolberg(Rheinl)Hbf Aachen-Rothe Erde;1;a
8000085;8001584;9511014; D sseldorf-Benrath D sseldorf Hbf;8000082;9510136; D sseldorf Flughafen D sseldorf Hbf;1;a
8000178;8000084;9510139; D ren Horrem;8000208;9510136; K ln-Ehrenfeld Horrem;1;a
8003553;8001886;9510137; Eschweiler Hbf Langerwehe;8000084;9510136; D ren Langerwehe;1;a
8001584;8006713;9510139; Leverkusen Mitte D sseldorf-Benrath;8000085;9510136; D sseldorf Hbf D sseldorf-Benrath;1;a
8003368;8000207;9212850; K ln Hbf K ln Messe/Deutz;8000209;9510136; K ln-M lheim K ln Messe/Deutz;1;a
8000208;8000178;9510139; Horrem K ln-Ehrenfeld;8000207;9510136; K ln Hbf K ln-Ehrenfeld;1;a
8000209;8003368;9510137; K ln Messe/Deutz K ln-M lheim;8006713;9510136; Leverkusen Mitte K ln-M lheim;1;a
8006713;8000209;9510135; K ln-M lheim Leverkusen Mitte;8001584;9510136; D sseldorf-Benrath Leverkusen Mitte;1;a
8000348;8000406;9510139; Aachen-Rothe Erde Stolberg(Rheinl)Hbf;8001886;9510136; Eschweiler Hbf Stolberg(Rheinl)Hbf;1;a
8001886;8000348;9510139; Stolberg(Rheinl)Hbf Eschweiler Hbf;8003553;9510136; Langerwehe Eschweiler Hbf;1;a
```

- j : n ud correspondant   un carrefour ou un arr t de correspondance ;
- i : n ud origine ;
- ligne i : num ro de ligne du tron on origine ;
- textei : libell  du tron on origine ;
- k : num ro de n ud destination ;
- lignek : num ro de ligne du tron on destination ;
- textk : libell  du tron on destination ;
- volau : volume des flux de mouvement tournant ou de correspondance ligne   ligne.

Seuls les mouvements tournants ou volumes de correspondance ligne   ligne, dont le volume de flux est non nul, apparaissent dans le fichier.

## 7 - Exemples d'utilisation

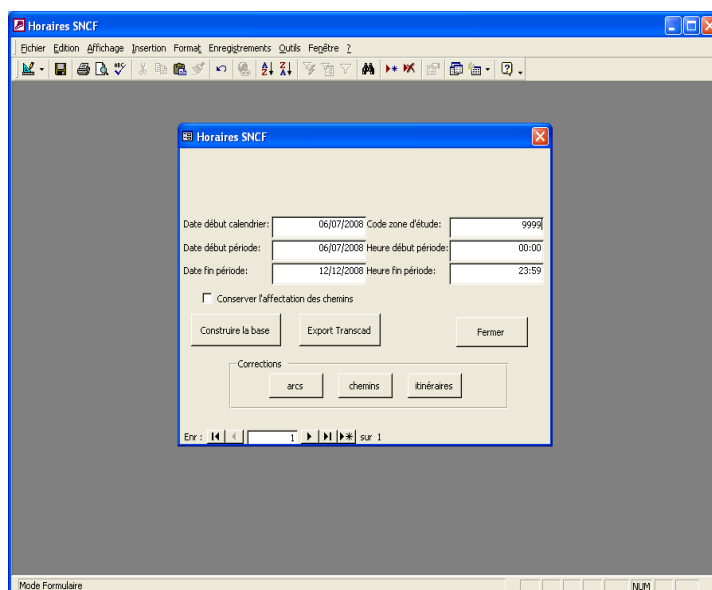
Cette partie fournit quelques exemples variés d'utilisation de l'outil MUSLIW. Pour chacune des questions traitées, les étapes du traitement du calcul avec MUSLIW sont détaillées.

### 7.1 - Comment générer le réseau TC avec BDFER ?

Dans l'outil BDFER [Construction d'un SIG du transport ferroviaire national de voyageurs sous TransCAD - CETE de Nord-Picardie - Sétra - février 2008], développé par le CETE Nord-Picardie, lors de l' « Export Transcad », l'outil crée un fichier « sncf\_MUSLIW.txt ».

Ce fichier est un « réseau MUSLIW ».

Attention, le réseau ne sera créé que pour les trains ayant au moins un point d'arrêt dans la zone sélectionnée, pendant la période sélectionnée et durant la plage horaire choisie.



Il faut donc veiller à effectuer la bonne sélection avant de lancer la construction de la base. L'utilisateur peut même être amené à changer le code zone des noeuds ferres sous TransCAD ou MapInfo, en sachant que les gares pour lesquelles le code zone serait égal à zéro seront ignorées.

Pour cela, il suffit d'ouvrir ou d'importer la table MapInfo « noeuds.tab ». Le lancement de la construction de la base avec un code région « 9999 » prendra en compte toutes les gares sauf celles dont le code est « 0 ».

Dans l'outil BDFER la codification des gares parisiennes de RER et grandes lignes sont différentes. Pour permettre la correspondance via le RER dans Paris, il est nécessaire d'ajouter des liens piétons entre les gares RER et grandes lignes des grandes gares parisiennes (Gare du Nord, Gare de Lyon, Gare Montparnasse, Gare de l'Est, Gare St-Lazare, Gare d'Austerlitz, Marne la Vallée Chessy, Roissy Charles de Gaulle, Massy, ...).

## **7.2 - Comment connecter un réseau à horaires avec un réseau de transports individuels ?**

### **7.2.1 -Dans TransCAD**

Les arrêts transports collectifs se présentent sous forme d'une couche d'objets ponctuels et il est donc nécessaire d'utiliser la commande « *Tools>Map Editing>Connect* » pour les connecter au réseau linéaire. La connexion se fait, soit par projection perpendiculaire du point représentant l'arrêt sur le tronçon du réseau linéaire de transport individuel, soit par connexion au nœud extrémité d'un tronçon du réseau de transport individuel le plus proche.

Une attention particulière doit être apportée à la gestion des numéros de nœuds. Les numéros de nœuds des arrêts TC doivent en effet être différents de ceux du réseau de transport individuel.

Les tronçons de transport en commun sur lesquels on peut se déplacer à pied ou en voiture particulière doivent être sélectionnés dans le réseau de transport individuel puisqu'ils peuvent à la fois être parcourus en transports individuel et collectif par la description des horaires des différents services.

La procédure pour générer le fichier réseau à partir d'une table TransCAD est décrite au § 3.4.

### **7.2.2 -Dans MapInfo**

Les utilisateurs ne disposant pas de TransCAD peuvent effectuer cette manipulation dans MapInfo.

Cependant, cette procédure n'est pas disponible dans la version standard du logiciel. Le CETE Nord-Picardie a donc développé l'utilitaire CONNECT.MBX. Celui-ci connecte une couche d'objets ponctuels à une couche d'objets linéaires en projetant perpendiculairement chaque point sur l'objet linéaire le plus proche, qui est alors scindé en deux parties.

MapInfo ne permet pas de faire la codification automatique en standard des numéros de nœud. Pour codifier son réseau sous Mapinfo, l'utilisateur peut se servir de l'outil « RESEAUX.MBX » développé par le CETE Nord-Picardie qui permet de définir les numéros de nœuds initiaux et finaux des tronçons, générer une table des nœuds et exporter le réseau au format MUSLIW.

### **7.2.3 -Comment calculer un accès avec des déplacements se déroulant sur plusieurs jours avec des périodes répétitives ?**

Dans les réseaux urbains, les services se répètent toutes les semaines. Ainsi l'offre d'un lundi de période scolaire sera souvent la même quelle que soit la semaine. L'identification de la circulation de chaque service peut donc se faire généralement par un calendrier défini par une semaine type.

Le calendrier a alors une forme sur sept caractères de type :

- « OOOOONN » pour les jours ouvrables, ;
- « NNNNNOO » pour le week-end.

Pour prendre en compte dans l'accessibilité un déplacement se déroulant sur plusieurs jours (par exemple un déplacement « à cheval » entre dimanche et lundi), un tel calendrier peut poser problème car le lundi (jour 0) est le premier jour du calendrier et le dimanche (jour 6) le dernier jour.

Pour permettre ce type de calcul, il suffit de répéter pour chaque service le calendrier :

- « OOOOONNOOOOONN » pour les jours ouvrables ;
- « NNNNNOONNNNNOO » pour le week-end ;

puis de rentrer comme jour d'arrivée le lundi (jour 7) et/ou le dimanche comme jour de départ (toujours jour 6) ; le module autorisera le déplacement « à cheval » entre le dimanche et le lundi.

## 7.3 - Comment calculer l'accessibilité aller-retour en restant un temps défini à destination ?

Pour certaines études, l'accessibilité n'a pas besoin d'être forcément caractérisée par des temps de parcours, mais par des possibilités de desserte de type :

- allers-retours sur la journée en restant X heures à tel endroit ;
- allers-retours sur une demi-journée ;
- allers-retours sur la semaine.

Ces types d'indicateurs sont particulièrement adaptés à des offres peu denses par lignes d'autocars, aériennes, voire ferroviaires.

La procédure de calcul de ces indicateurs est la suivante :

- construire un réseau MUSLIW de « m » tronçons dont le plus haut numéro de nœud est « n ». Le réseau peut contenir à la fois des tronçons à horaires et des tronçons de transport individuel ou ne comporter que des tronçons à horaires. La desserte sera alors estimée par rapport à une gare ou un arrêt de transport en commun ;
- dupliquer ce réseau en modifiant tous les numéros de nœuds et en ajoutant à ceux du réseau initial une valeur ronde de préférence afin de pouvoir effectuer rapidement la correspondance. Aucun numéro de nœud ne doit être commun aux deux réseaux.

Soit k, le nombre qui ajouté au numéro de nœud du réseau original donne le numéro du nœud correspondant dans le réseau dupliqué.

Le nœud p du réseau original aura le numéro p+k dans le réseau dupliqué.

- déterminer le nœud i dans le réseau original comme étant le lieu de destination où doivent se trouver les usagers pendant une période h avant de pouvoir revenir. Son équivalent dans le réseau dupliqué sera alors le nœud i+k ;
- fusionner les deux réseaux en ajoutant le réseau dupliqué au réseau original. Pour ce faire, exécuter le mode console (Windows\Exécuter\cmd) puis se positionner dans le bon répertoire et entrer la commande « copy reseau1.txt+reseau2.txt reseau\_total.txt » ;
- ajouter un tronçon de type mode individuel :
  - nœud origine : i ;
  - nœud destination : i+k ;
  - temps : temps durant lequel l'utilisateur doit rester à destination ;
  - longueur : 0 ;



—1 pour les autres champs, excepté le champ texte où l'on peut saisir le lieu de destination par exemple.

•créer un fichier matrice :

—nœud origine : i ;

—nœud destination : i+k ;

—volume : indifférent ;

—jour : le jour de la période sur laquelle l'analyse doit porter ;

—heure : Heure de départ ou d'arrivée souhaitée ;

—type d'heure : « a » ou « d » selon que vous voulez fixer un jour et une heure de retour limites ou un jour et une heure de départ limites ;

—vous pouvez créer plusieurs lignes pour spécifier plusieurs heures de départ ou d'arrivée possibles ;

•lancer l'affectation à horaires dans MUSLIW en fixant bien le nombre de jours souhaité pour la recherche des plus courts chemins ;

•analyser les fichiers résultats.

## 7.4 - Comment prendre en compte l'accessibilité voiture + transports collectifs + marche à pied?

Dans un premier temps, il s'agit d'effectuer les manipulations présentées au paragraphe 5.2 pour construire un réseau multimodal transports-collectifs+marche à pied. Parallèlement, pour le réseau VP, il s'agit de remplir (comme pour le réseau de marche à pied) le champ temps par une estimation du temps de parcours sur chaque tronçon en voiture, puis de modifier les numéros de nœud pour qu'aucun ne soit commun avec le réseau TC+marche à pied. Il s'agit ensuite de connecter les gares, pôles d'échanges ou parcs relais au réseau voiture par la méthode proposée.

**NB :** Si vous souhaitez que le mode voiture ne soit utilisable qu'au départ du déplacement et interdire aux usagers de pouvoir ensuite utiliser encore la voiture en arrivant à destination, vous ne devez connecter les deux réseaux que dans un sens :

•sens VP -> Gare pour une utilisation de la voiture uniquement en rabattement ;

•sens Gare -> VP pour une utilisation de la voiture uniquement en diffusion.

Le temps à affecter au connecteur doit correspondre au temps supplémentaire que l'on souhaite ajouter au temps de correspondance par défaut pour tenir compte des délais de précaution de l'usager à son arrivée en gare.

Pour du rabattement, le fichier MUSLIW doit indiquer une origine sur le réseau VP et une destination sur le réseau TC+marche à pied, et inversement pour de la diffusion en voiture.

Attention, le coefficient de pondération du poids de marche à pied sera également utilisé pour pondérer le temps VP. Pour assurer des rabattements cohérents, la valeur de ce coefficient devrait être comprise entre 1.5 et 3, de l'ordre de celui utilisé pour la marche à pied. Plus la valeur de la pondération sera forte, plus l'usager aura tendance à se rabattre vers une gare plus proche, même si elle est moins bien desservie. A l'inverse un coefficient de « 1 » privilégiera dans la plupart des cas un déplacement uniquement en voiture.

## **7.5 - Comment prendre en compte des réseaux vélos en libre service en imposant les points de retrait et dépose des vélos ?**

Le principe est le même que précédemment. Il faut dupliquer le réseau marche à pied en sélectionnant uniquement les tronçons cyclables et modifiant les numéros de nœuds.

Puis, il est nécessaire d'introduire des temps vélo et connecter les points de retrait des vélos en libre service avec les points correspondants dans le réseau marche à pied. Le champ temps sur les connecteurs doit correspondre au temps moyen pour prendre un vélo (sens marche à pied-vélo) et pour le rendre (sens vélo-marche à pied).

La pondération étant la même pour la marche à pied que pour le vélo, le vélo libre service sera systématiquement utilisé si la prise, dépose, circulation à vélo est plus rapide, ainsi que le trajet à pied, pour se rendre à une borne et entre la borne et sa destination. Il faut donc rentrer comme nœuds origine et destination deux nœuds du réseau marche à pied. Avec ce paramétrage, l'utilisateur retirera un vélo uniquement si cela lui fait gagner du temps et il sera contraint de reposer le vélo emprunté à une borne.

## **7.6 - Comment calculer des indicateurs d'accessibilité sur une période ?**

MUSLIW calcule le plus court chemin pour une date et une heure données. Si l'on souhaite bâtir des indicateurs d'accessibilité pour tenir compte de la fréquence de la desserte, par exemple, ou ne retenir que les temps de trajet les plus courts, il est nécessaire de répéter la même origine-destination dans le fichier matrice mais avec des jours et/ou des heures différentes (par pas de 1h, 30min, 15min, 5 min ou 1min).

Un traitement statistique des fichiers résultats permettra ensuite de calculer l'indicateur adéquat pour chaque étude (moyenne, minimum, maximum, écart-type, quantile, etc.). Le fichier des résultats d'affectation prend alors tout son intérêt pour analyser les différents itinéraires et leurs poids respectifs.

## **7.7 - Quelles sont les données nécessaires pour le dessin d'isochrones ?**

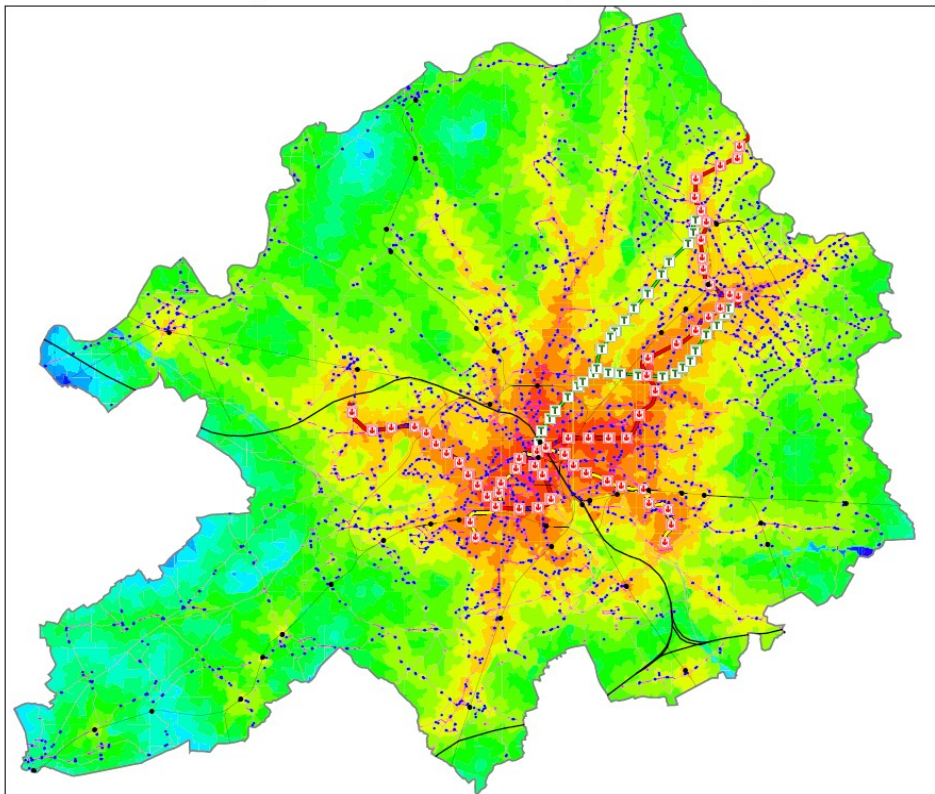
Pour une définition des notions d'accessibilité et des applications qui peuvent en être faites, l'utilisateur pourra consulter le rapport d'études du Sétra « Accessibilité des territoires et des services - Notions et représentation » - juin 2008<sup>2</sup>.

Mis à part le cas particulier des « Drive-in », on accède toujours à son activité à pied. C'est pourquoi, seul le réseau piéton ou voiture est nécessaire pour la réalisation d'isochrones.

---

2 téléchargeable sur les sites Internet et Intranet du Sétra sous le lien :  
<http://www.setra.developpement-durable.gouv.fr/Accessibilite-des-territoires-et.html>

Accessibilité en transports en commun et marche à la Gare Lille Europe  
Arrivée pour 9h - vitesse de marche 4km/h - Horaires Printemps 2008



*Illustration 2 : Isochrones TCU+marche construite sous SIG à partir des temps sur le réseau piéton*

Dans le cas de la représentation du réseau TC seul, il faut au minimum une géolocalisation des arrêts pour avoir des pastilles de différentes couleurs.

Il faut ensuite récupérer les résultats des temps par tronçon et les associer aux tronçons équivalents sous SIG, en général via les numéros de nœuds début et fin.

### **7.7.1 -Dans TransCAD**

TransCAD propose des fonctionnalités pour réaliser des isochrones à partir de temps de parcours sur le réseau routier. La procédure est simple, automatique et documentée dans le guide Sétra d'utilisation de TransCAD pour la représentation de l'offre et de la demande de transport multimodales - Version 2.3 d'avril 2010 .

### **7.7.2 -Dans MapInfo**

MapInfo dans sa version utilisée dans le Ministère, ne permet pas ce type d'analyse. Le CETE Nord-Picardie a développé un utilitaire « RESEAUX.MBX » qui permet de réaliser ce type d'analyse et de générer un

graphe (numéro de nœuds) à partir d'une couche de points. Cet utilitaire peut être mis à disposition des utilisateurs qui le souhaitent.

## 8 - Simplification d'un réseau

La procédure de simplification de réseau permet de supprimer tous les nœuds intermédiaires qui ne correspondent pas à des carrefours, c'est-à-dire qui ne sont connectés qu'à deux nœuds et qui ne sont pas des tronçons d'injection. Cette procédure permet de diminuer la taille du réseau. Les données de temps et de longueur sont la somme de celles des tronçons fusionnés.

Pour effectuer cette procédure, il faut au préalable importer un réseau puis lancer la simplification via le menu « Réseaux>Simplifier ». L'outil simplifie le réseau et l'exporte sous forme de fichier réseau Emme/2 sous la racine « c:\d211.in ».

## 9 - Visualisation

### 9.1 - Introduction

Des fonctionnalités de visualisation sont disponibles dans MUSLIW. Il s'agit d'outils de contrôle et non de cartographie. Leur utilisation peut être utile pour vérifier certaines manipulations ou calculs, mais il est conseillé d'exporter les résultats de calculs d'accessibilité avec MUSLIW sous des logiciels SIG pour cartographier et visualiser les résultats.

Afin de visualiser un réseau, il est nécessaire que MUSLIW puisse déterminer les coordonnées des nœuds.

Cela est possible avec :

- un réseau Emme/2 ;
- un réseau multimodal (si le réseau contient l'extension permettant l'importation des coordonnées des nœuds) après une affectation TC à horaires.

### 9.2 - Réseau multimodal à horaires étendu

Le format du réseau multimodal à horaires étendu est tel que le fichier comprend deux sections précédées comme dans le format Emme/2 par une commande « t nodes » et « t links ».

#### 9.2.1 -Section « t nodes »

Elle comprend 4 champs :

- i : numéro de nœud ;
- x : abscisse du nœud ;
- y : ordonnée du nœud ;
- texte : libellé du nœud (facultatif).

t nodes

35;497303.56;2684361.06;"ASHFORD"  
 152;1006003.11;2411812.07;"KEHL"  
 199;938537.32;2481957.03;"SARREBRUCKEN HBF"  
 283;874816.62;2518401.05;"LUXEMBOURG"  
 286;872892.00;2508110.26;"BETTEMBOURG"  
 357;1023751.33;1879717.16;"VINTIMILLE"  
 432;773961.05;2824872.20;"AMSTERDAM CENTRAL"  
 434;761850.58;2759128.70;"DORDRECHT"  
 436;738594.77;2789192.92;"LA HAYE"  
 445;747984.26;2729727.48;"ROOSENDAAL"  
 446;747984.26;2773544.15;"ROTTERDAM CENTRAL"  
 3541;767701.54;2817360.82;"SCHIPOL"  
 450;996040.10;2296314.44;"BALE"  
 456;951650.02;2234222.78;"NEUFCHATEL"

MUSLIW n'étant pas un SIG, l'utilisateur n'a pas besoin de renseigner les coordonnées dans un système de projection particulier.

### 9.2.2 -Section « t links »

Une ligne « t links » précède la section dont le format est identique à celui décrit au 6.1, c'est-à-dire au format classique du réseau multimodal MUSLIW.

t links

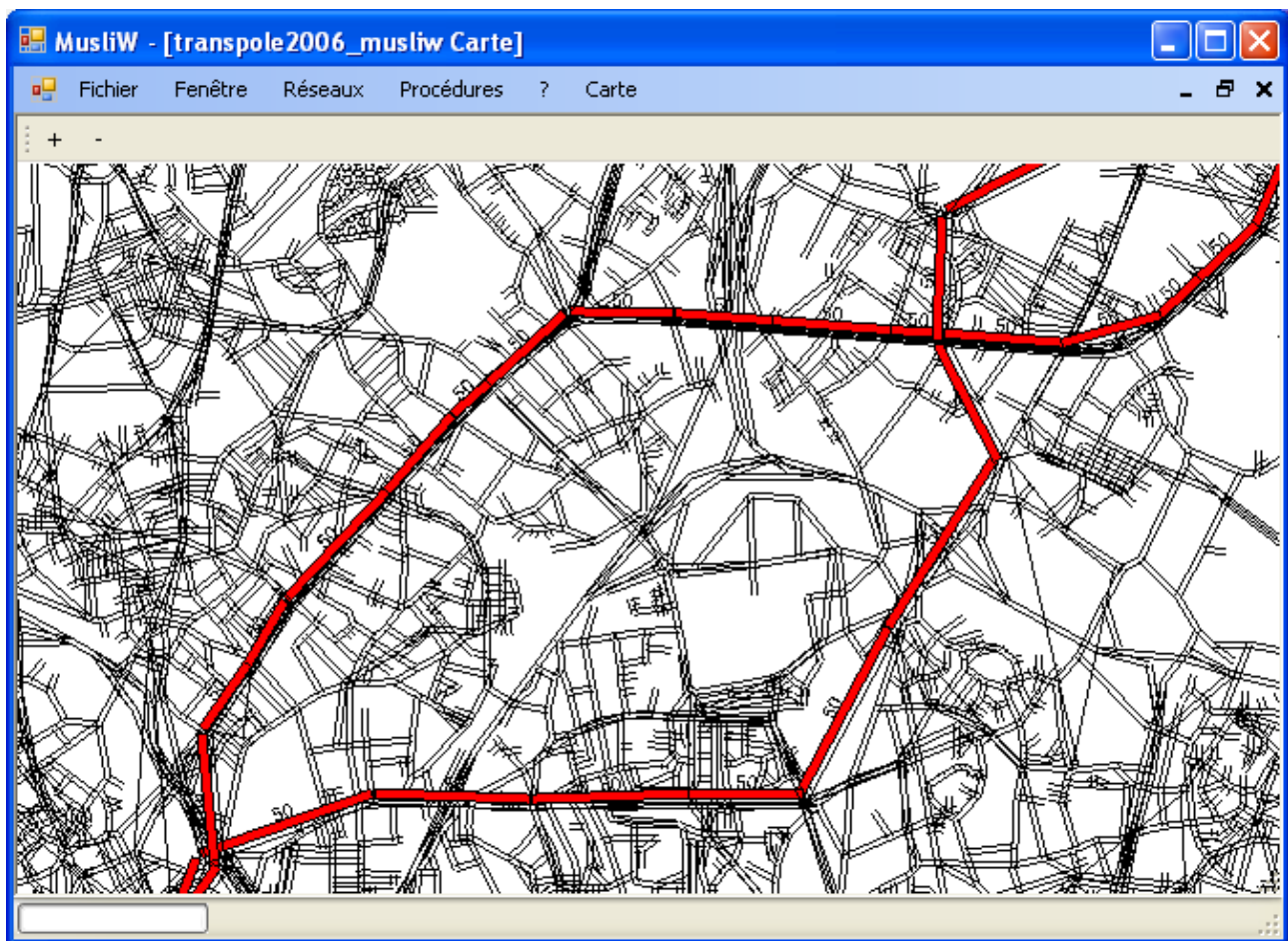
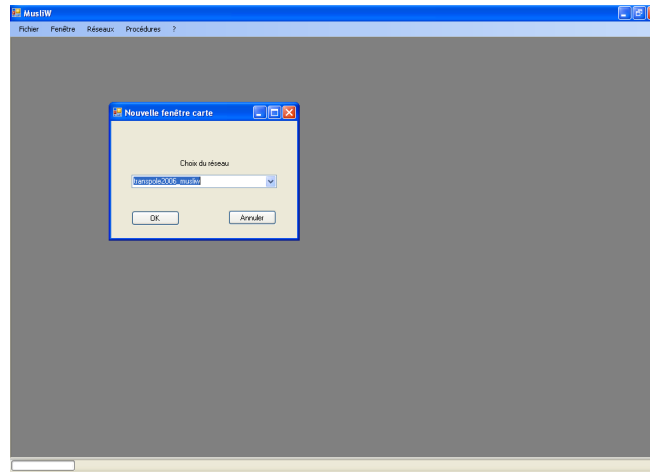
34;1473;-1;-  
 1;770;8173;366.00;370.00;ONNNNNNONNNNNNONONNNNONONNNNONNNNNNNNONNNNN  
 34;1473;-1;-  
 1;770;8177;546.00;550.00;ONNNNNNOONNNNNNOONONNNNOONONNNNOONNNNNNOONNNNNNOONNNN  
 34;1473;-1;-1;770;8180;606.00;610.00;OO  
 34;1473;-1;-1;770;8184;786.00;790.00;OO  
 34;1473;-1;-1;770;8198;1206.00;1210.00;OO  
 34;1473;-1;-1;771;8170;366.00;370.00;NOOOOOONOOOOOONONOOOONONOOOONOOOOONOOOOOONOOOOOONOO  
 34;1473;-1;-1;771;8182;726.00;730.00;OO  
 34;1473;-1;-1;771;8187;846.00;850.00;OO  
 34;1473;-1;-1;772;8175;486.00;490.00;OO  
 34;1473;-1;-1;772;8191;966.00;970.00;OO  
 34;1473;-1;-1;772;8194;1025.00;1029.00;OO  
 34;1473;-1;-1;772;8196;1085.00;1089.00;OO  
 35;36;-1;-1;4;1750;1304.00;1333.00;OOOOOOOOOOOOOOOONNOOONOOOOOOOOOOOOOOOOOOOOOOOOOO  
 35;36;-1;-  
 1;4;1749;1304.00;1332.00;NN  
 35;36;-1;-  
 1;5;1701;668.00;698.00;NN

## 9.3 - Affichage d'une carte

L'affichage d'une fenêtre carte s'effectue via le menu « Fenêtre>Carte ». L'utilisateur doit sélectionner le réseau qu'il souhaite afficher.

L'utilisateur peut naviguer sur cette affichage de plusieurs façons :

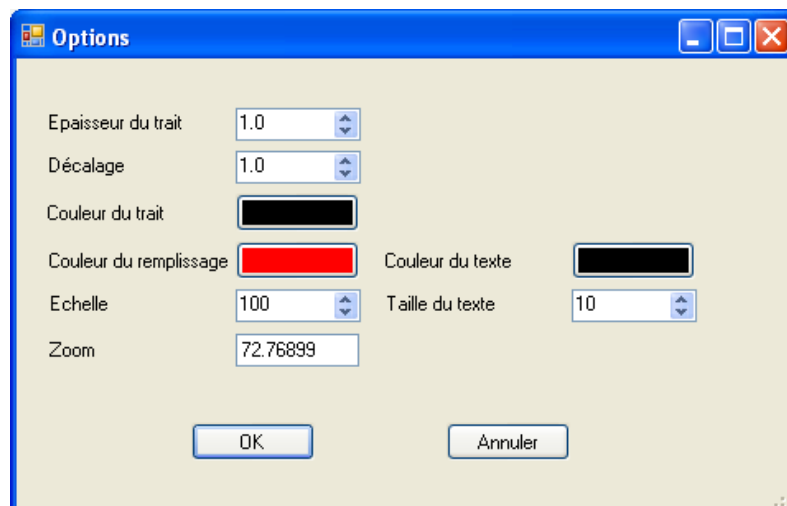
- se déplacer par cliquer-déplacer ;
- zoomer d'un facteur 2 avec le bouton « + » ;
- dézoomer d'un facteur 0,5 avec le bouton « - ».





MUSLIW affiche en épaisseurs proportionnelles les trafics affectés et écrit la valeur des trafics si l'encombrement de l'étiquette n'excède pas la longueur de l'arc.

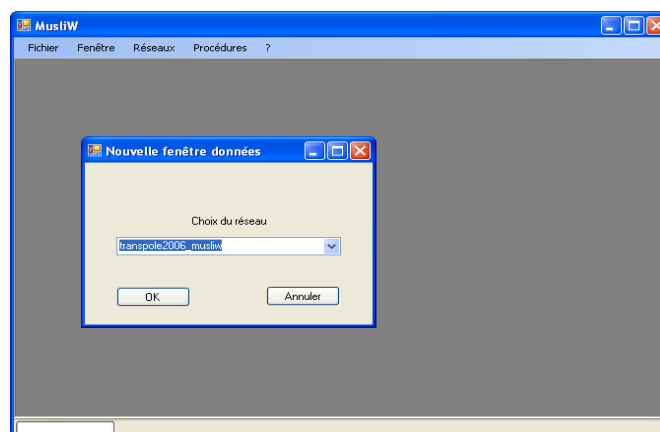
Quelques options de paramétrage de l'affichage sont disponibles via le menu « Carte>Options » :



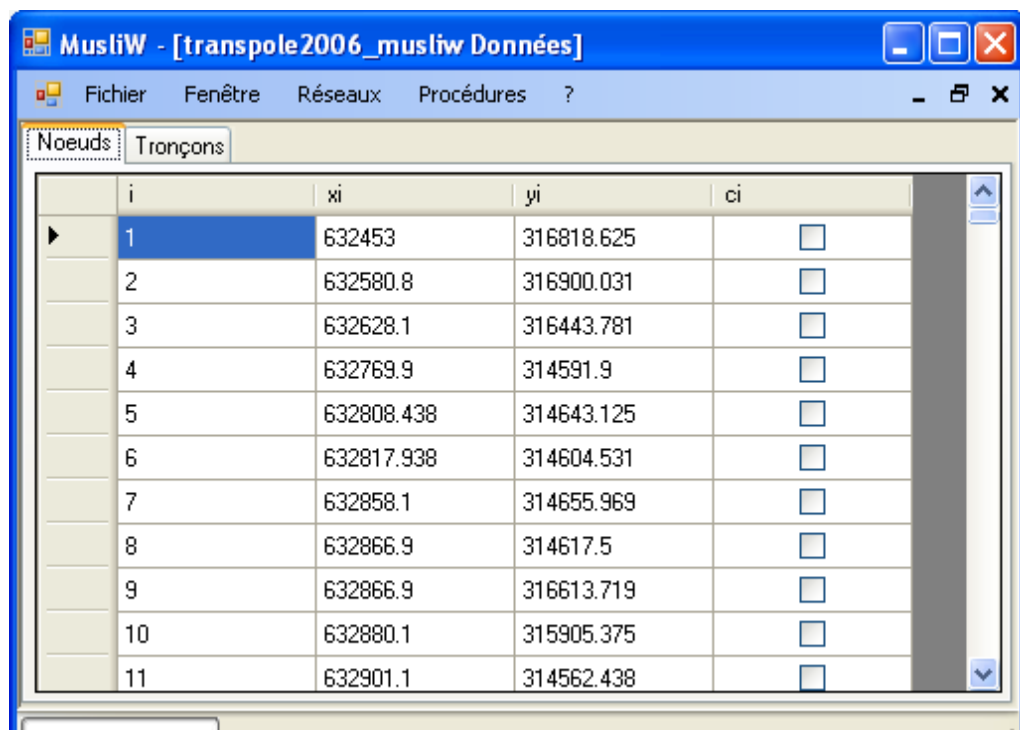
- épaisseur du trait : épaisseur en pixel de l'arc ;
- décalage : décalage en pixels d'un sens par rapport à l'autre ;
- couleur du trait : couleur de l'arc ;
- couleur de remplissage : couleur du remplissage pour les épaisseurs proportionnelles ;
- échelle : facteur d'échelle pour les épaisseurs proportionnelles (plus l'échelle sera petite, plus les épaisseurs seront importantes, relativement parlant) ;
- taille du texte : taille de la police ;
- zoom : facteur de zoom.

## 9.4 - Affichage des données

L'affichage d'une fenêtre de données s'effectue via le menu « Fenêtre>Données ». L'utilisateur doit alors sélectionner le réseau qu'il souhaite afficher. Cette fonctionnalité n'est pas disponible sous Linux



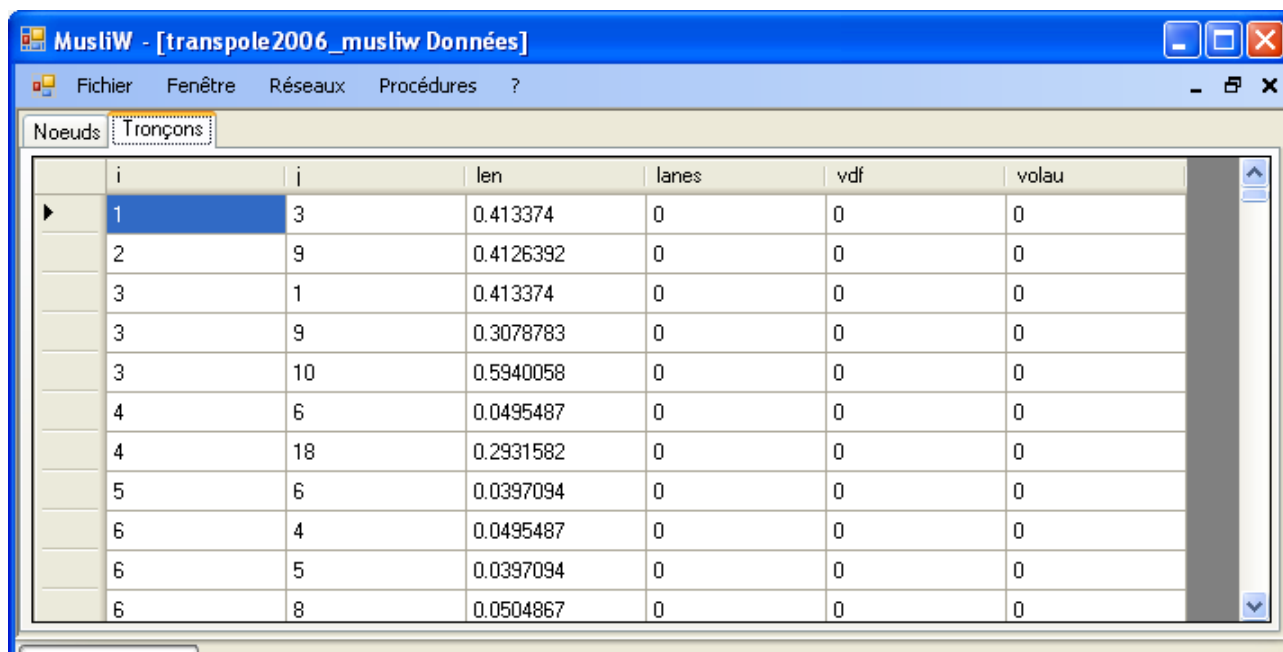
L'affichage des données est limité à certaines d'entre elles.



	i	xi	yi	ci
▶	1	632453	316818.625	<input type="checkbox"/>
	2	632580.8	316900.031	<input type="checkbox"/>
	3	632628.1	316443.781	<input type="checkbox"/>
	4	632769.9	314591.9	<input type="checkbox"/>
	5	632808.438	314643.125	<input type="checkbox"/>
	6	632817.938	314604.531	<input type="checkbox"/>
	7	632858.1	314655.969	<input type="checkbox"/>
	8	632866.9	314617.5	<input type="checkbox"/>
	9	632866.9	316613.719	<input type="checkbox"/>
	10	632880.1	315905.375	<input type="checkbox"/>
	11	632901.1	314562.438	<input type="checkbox"/>

Illustration 4 : Onglet noeuds

Illustration 5 : Onglet tronçons



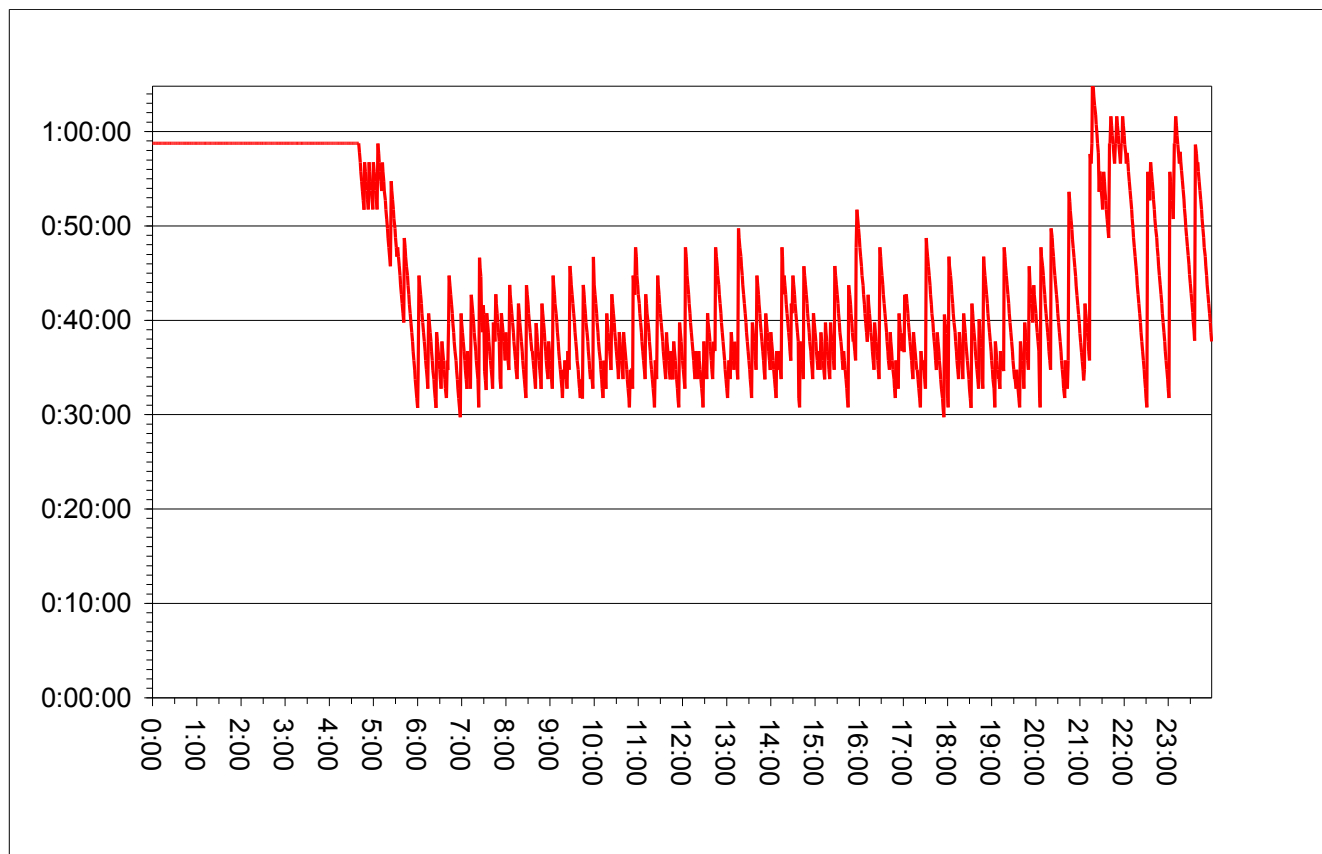
	i	j	len	lanes	vdf	volau
▶	1	3	0.413374	0	0	0
	2	9	0.4126392	0	0	0
	3	1	0.413374	0	0	0
	3	9	0.3078783	0	0	0
	3	10	0.5940058	0	0	0
	4	6	0.0495487	0	0	0
	4	18	0.2931582	0	0	0
	5	6	0.0397094	0	0	0
	6	4	0.0495487	0	0	0
	6	5	0.0397094	0	0	0
	6	8	0.0504867	0	0	0

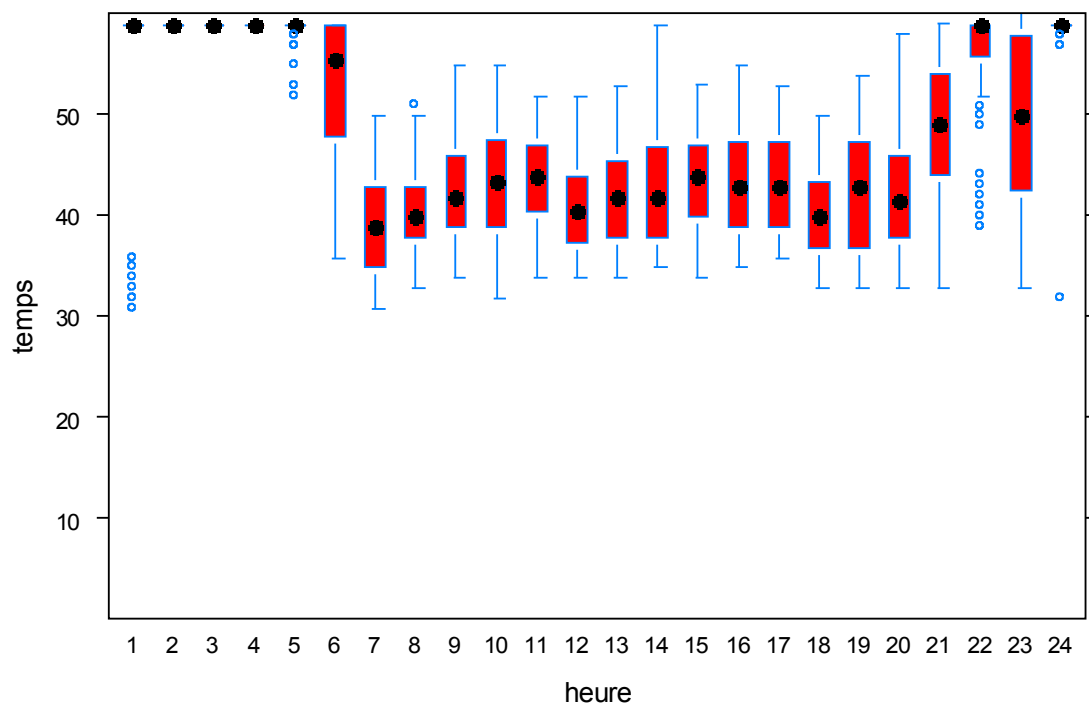


# 10 - Exploiter les résultats

## 10.1 - Exploitations statistiques

Une exploitation statistique des calculs de MUSLIW est possible, moyennant le recours à des outils ad hoc. Une illustration de traitements réalisables est fournie ci-dessous, le premier graphique représentant les variations de temps de parcours sur un couple origine-destination donné en fonction de l'horaire de départ, le second transcrivant cette donnée sous la forme valeur moyenne, écart type et intervalle de confiance par tranche horaire,





Pour l'exploitation des résultats issus de calcul sur les réseaux et les matrices importants, le CETE Nord-Picardie recommande le recours à des logiciels tels que R<sup>3</sup>, permettant de traiter des fichiers conséquents et de réaliser des graphiques de haut niveau, en seulement quelques lignes de script.

Ces logiciels peuvent également être très efficaces pour construire les fichiers réseau et matrice et préparer les fichiers pour une utilisation sous SIG.

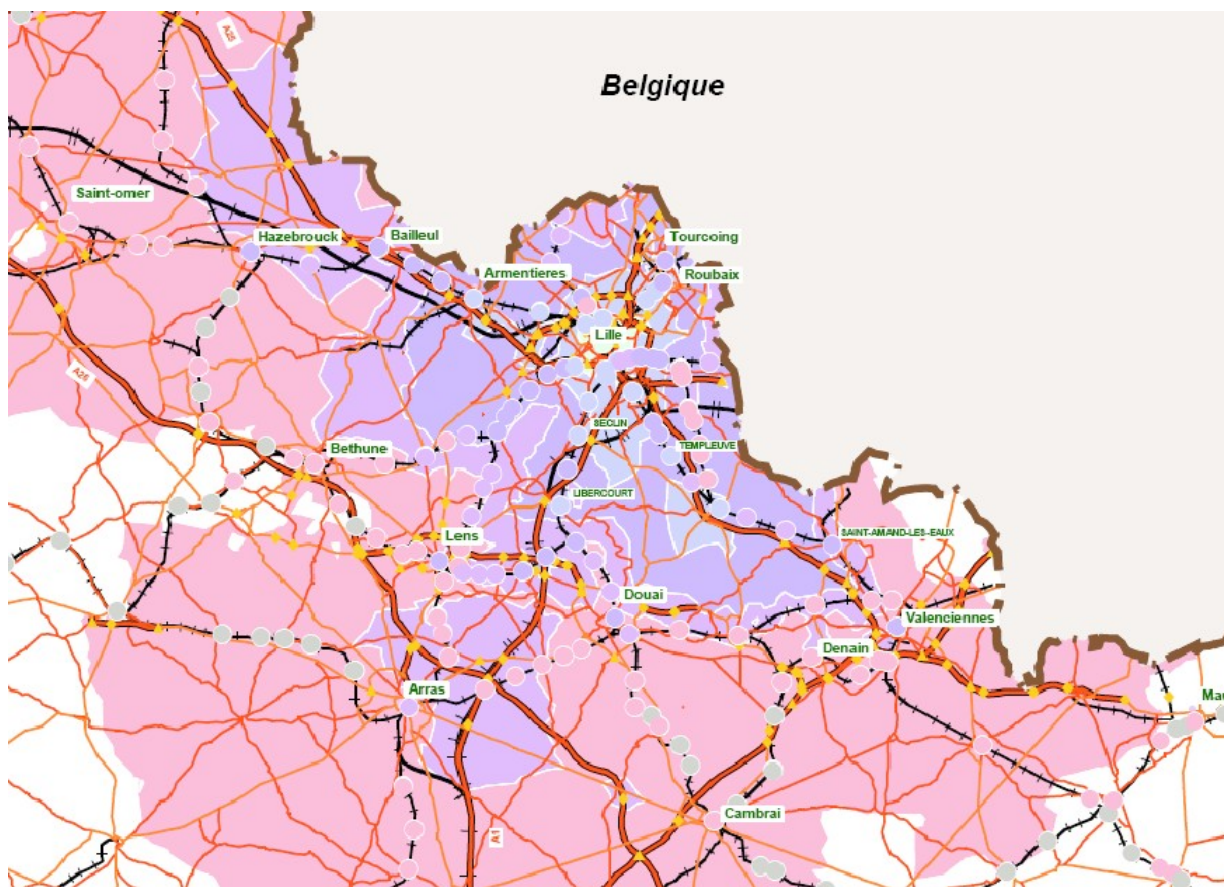
Les packages (memisc, survey, voire Hmisc) sont très utiles pour les traitements de données et (ggplot2, lattice) pour les représentations graphiques.

---

<sup>3</sup> R Development Core Team (2009). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.

## 10.2 - Exploitations cartographiques

### 10.2.1 -TransCAD



TransCAD permet en standard de visualiser des isochrones. Pour cela il faut effectuer une jointure entre un fichier de résultats issu de MUSLIW et une table TransCAD.

### **10.2.2 -Mapinfo**

Dans Mapinfo, le CETE met à disposition l'outil RESEAUX.MBX qu'il a développé. Pour mémoire, ce dernier permet :

- de générer des graphes automatiquement à partir de couches d'objets linéaires ;
- de dessiner des surfaces isochrones à partir de temps sur des points ou des tronçons ;
- de visualiser les itinéraires.

# 11 - Glossaire

## **Transport Individuel**

Les transports individuels désignent dans cette étude les modes de transports suivants :

- la Marche à Pied (MAP) ;
- la Voiture Particulière (VP) ;
- les vélos en libre service de type Vélib' ou Vélov.

## **Réseau de Transport Individuel (ou réseau TI)**

Le réseau de transport individuel désigne l'ensemble des tronçons sur lesquels s'effectuent les déplacements par ce mode. Dans le codage du réseau pour MUSLIW, pour une même infrastructure utilisable à pied ou en voiture particulière, il faudra coder deux tronçons distincts, avec des nœuds origine et destination différents reliés entre eux par des arcs fictifs.

## **Transports collectifs**

Les transports collectifs désignent dans cette étude tous les modes de transport qui se caractérisent par une offre de services pré-déterminée basée sur des itinéraires et des arrêts précis.

## **Réseau de Transports Collectifs (ou réseau TC)**

Le réseau de transport collectif désigne l'ensemble des tronçons sur lesquels les services de TC sont opérés. Sur un même tronçon peuvent être opérés des services de TC distincts.

## **Algorithme de recherche de plus courts chemins en stratégies optimales**

L'algorithme de recherche de plus courts chemins en stratégies optimales permet de trouver, pour un couple origine-destination donné, l'ensemble des chemins intéressants pour un usager des TC, en tenant compte de la fréquence des différents services qui s'offrent à lui, lorsqu'il arrive à un arrêt desservi par plusieurs lignes. Pour une description détaillée de l'algorithme, on renvoie le lecteur au guide du Certu de 2003 (Modélisation des déplacements urbains de voyageurs).

# Bibliographie

- «Accessibilité des territoires et des services - Notions et représentation» - Sétra 6 juin 2008.  
téléchargeable sur le site Internet du Sétra sous le lien : <http://www.setra.developpement-durable.gouv.fr/Accessibilite-des-territoires-et.html>
- « Guide d'utilisation de TransCAD pour la représentation de l'offre et de la demande de transport multimodales (Version 2.3) » - Sétra – avril 2010.
- « Construction d'un SIG du transport ferroviaire national de voyageurs sous TransCAD » - CETE de Nord-Picardie, Sétra - février 2008.
- « Modélisation des déplacements urbains de voyageurs » - Certu – 2003.
- « Shortest Path Algorithms : An Evaluation Using Real Road Networks » - Zhan, F. B., and Noon, C. E. in Transportation Science – 1996.

Les études d'accessibilité font appel à un nombre important de données qu'il faut traiter informatiquement avec des outils adaptés. Les logiciels de planification des transports proposent de tels outils mais ils peuvent être onéreux, parfois difficiles à mettre en œuvre et nécessitent généralement un investissement humain important (prise en main du logiciel, codification des données, etc.). Par ailleurs, les algorithmes permettant le calcul d'accessibilité sont propres à chaque logiciel et peuvent être inadaptés à certains cas (accessibilité multimodale, prise en compte de fiches horaires dans les TC, codage de systèmes de vélo en libre service, temps de parcours routiers variables selon l'heure et le jour, etc.).

Afin de permettre des calculs d'accessibilité multimodale à partir de données avec des formats largement répandus, le Cété Nord-Picardie a développé un utilitaire appelé MUSLIW.

Ce document décrit comment utiliser l'outil MUSLIW. Il présente les fondements théoriques de la recherche de plus court chemin sur un graphe et les principes retenus pour les algorithmes implémentés dans le logiciel. Il détaille le fonctionnement de l'utilitaire pour calculer des indicateurs d'accessibilité sur des réseaux multimodaux mêlant transport individuel et transport collectif. Il fournit également quelques exemples de codification et de paramétrage permettant de répondre à plusieurs types de questions liées à l'accessibilité. Enfin, il décrit les autres fonctionnalités de l'outil MUSLIW : simplification de réseau, affichage de cartes et de données.

---

#### Rédacteur

Patrick PALMIER – CETE Nord-Picardie

téléphone : 33 (0)3 20 49 60 70

mél : [patrick.palmier@developpement-durable.gouv.fr](mailto:patrick.palmier@developpement-durable.gouv.fr)

#### Service d'études sur les transports, les routes et leurs aménagements

46, avenue Aristide Briand – BP 100 – 92225 Bagneux Cedex – France

téléphone : 33 (0)1 46 11 31 31 – télécopie : 33 (0)1 46 11 31 69

Document consultable et téléchargeable sur les sites web du Sétra :

- Internet : <http://www.setra.developpement-durable.gouv.fr>
- Intranet (Réseau ministère) : <http://intra.setra.i2>

Ce document ne peut être vendu. La reproduction totale du document est libre de droits.

En cas de reproduction partielle, l'accord préalable du Sétra devra être demandé.

© 2010 Sétra – Référence : 1037w – ISRN : EQ-SETRA--10-ED17--FR

Le Sétra appartient  
au Réseau Scientifique  
et Technique  
du Meeddm

