

451567

MINISTÈRE DES TRANSPORTS DU QUÉBEC  
SERVICE DE LA CARTOGRAPHIE

MINISTÈRE DES TRANSPORTS  
CENTRE DE DOCUMENTATION  
700, BOUL. RENÉ-LÉVESQUE EST,  
21<sup>e</sup> ÉTAGE  
QUÉBEC (QUÉBEC) - CANADA  
G1R 5H1

**PROJET DE GÉNÉRALISATION**

Développement d'outils de généralisation de données cartographiques numériques pour la production de diverses séries cartographiques à petite échelle

**RAPPORT FINAL**

REÇU  
CENTRE DE DOCUMENTATION  
29 SEP 1997  
TRANSPORTS QUÉBEC

**PGI** PHOTOSUR GEOMAT  
INTERNATIONAL

3 JUIN 1997

WNP  
TR  
PT  
CRIT  
125

TABLE DES MATIÈRES

---

	<u>PAGE</u>
1. INTRODUCTION .....	1
2. BILAN GÉNÉRAL DU PROJET .....	2
2.1 REVUE DES TRAVAUX EFFECTUÉS.....	2
2.2 ÉCHÉANCIER.....	2
2.3 LES DIFFICULTÉS RENCONTRÉES .....	3
2.4 LES RÉSULTATS.....	3
3. RECOMMANDATIONS.....	4
3.1 POINTS À SURVEILLER RELATIVEMENT À L'EXPLOITATION DU LOGICIEL.....	4
3.1.1 Structuration des données .....	4
3.1.2 Détection des éléments trop rapprochés .....	4
3.2 PLAN D'ACTION SUGGÉRÉ SUITE AU PRÉSENT PROJET.....	4
3.3 PERSPECTIVES D'AMÉLIORATION FUTURE DU LOGICIEL GÉNÉRALISATION-MTQ.....	5
4. CONCLUSIONS.....	7

## 1. INTRODUCTION

---

L'objectif du présent document est de présenter brièvement les résultats et les conclusions du projet de "Développement d'outils de généralisation de données cartographiques numériques pour la production de diverses séries cartographiques à petite échelle". Il s'agit d'un mandat octroyé par le Ministère des Transports du Québec à PHOTOSUR GEOMAT INTERNATIONAL le 19 décembre 1996.

La section 2 présente le bilan général du projet, tandis que la section 3 présente des recommandations, à la lumière des résultats du projet.

## 2. BILAN GÉNÉRAL DU PROJET

---

Ce chapitre décrit de façon générale les travaux effectués, l'échéancier, les difficultés rencontrées ainsi que les résultats du projet.

### 2.1 REVUE DES TRAVAUX EFFECTUÉS

---

Les travaux réalisés ont suivi exactement le plan du projet, dont voici les principales étapes :

- établissement des spécifications avec le Ministère;
- obtention de données d'essai;
- développement d'outils de reclassification et de sélection dans l'environnement Arc/Info;
- intégration de l'algorithme de lissage de PGI à l'environnement Arc/Info (voir ci-dessous) et développement de fonctions de filtrage;
- expérimentation avec les données d'essai;
- livraison de la version 0,5 du logiciel;
- développement de la détection des éléments trop près;
- amélioration des outils de filtrage de PGI (voir ci-dessous);
- documentation et rapport;
- livraison finale.

Un fait important à souligner est qu'en parallèle avec ce mandat, PGI a investi ses propres fonds dans un projet interne de recherche et développement, pour y développer des outils de filtrage et de lissage des données, en langage C, et en faire l'intégration à l'environnement Arc/Info. PGI met à la disposition du MTQ la version exécutable de ces outils. Ces outils originaux fournis au MTQ et intégrés au logiciel Généralisation-MTQ augmentent de beaucoup sa fonctionnalité et ses possibilités.

### 2.2 ÉCHÉANCIER

---

L'échéancier du projet a été suivi, à peu de chose près, malgré les quelques difficultés rencontrées (voir en 2.3). Les travaux ont débuté en janvier 1997 et une première livraison du logiciel a été effectuée tel que prévu, à la fin mars 97. En cours de mandat, deux présentations du logiciel ont été faites, ce qui a permis de valider la conception, le logiciel développé et les résultats. Suite à un accord en ce sens, l'installation du logiciel a été reportée au début juin, soit en même temps que la formation et la livraison de la documentation. La livraison finale, qui était prévue vers la fin mai a donc été reportée de quelques jours seulement.

### 2.3 LES DIFFICULTÉS RENCONTRÉES

---

Aucune difficulté significative n'a été rencontrée du côté de PGI. Cependant, PGI devait recevoir des données en format Arc/Info et ces données devaient être structurées selon les spécifications établies dans le devis. Au lieu de cela, nous avons reçu des données (2 feuillets) en format DGN modifié pour Arc/Info ainsi que des données (2 feuillets) non modifié; ces données étaient codifiées selon le devis D du MRN. Cela a obligé PGI à effectuer certaines corrections aux données. On estime qu'un effort d'environ 4 jours a été fourni par PGI pour surmonter cette difficulté.

Un délai d'environ 10 jours s'est produit dans la réception des données d'essai provenant de la BNDT, mais malgré cela, l'équipe de PGI a su s'adapter afin de ne pas retarder l'échéancier.

### 2.4 LES RÉSULTATS

---

Les livrables finaux du projet sont :

- le code source des programmes développés dans l'environnement Arc/Info;
- le code exécutable des outils de filtrage et de lissage des lignes;
- le guide de l'utilisateur;
- le guide technique;
- le présent rapport;
- des tracés des résultats d'expérimentation aux échelles de 1:50,000, 1:125,000, 1:500,000 et de 1:1,000,000, à partir des données sources 1:20,000 du MRN;
- des tracés des résultats d'expérimentation à l'échelle de 1:500,000, à partir des données sources 1:50,000 de la BNDT;
- des tracés des résultats d'expérimentation de détection des éléments trop rapprochés à l'échelle de 1:250,000, à partir des données sources 1:50,000 de la BNDT.

Une formation d'une journée a également été donnée aux utilisateurs du logiciel du Ministère au début juin 1997.

### 3. RECOMMANDATIONS

---

Ce chapitre présente certaines recommandations suite à l'expérimentation du logiciel avec les données d'essais.

#### 3.1 POINTS À SURVEILLER RELATIVEMENT À L'EXPLOITATION DU LOGICIEL

---

Les points particuliers suivants devront être surveillés par ceux qui exploiteront le logiciel au Ministère.

##### 3.1.1 STRUCTURATION DES DONNÉES

---

Les données à généraliser doivent être structurées en respectant au minimum les points suivants:

- les noeuds doivent être existants à tous les points d'intersection;
- le réseau routier doit être continu (sans interruption aux jonctions de feuillets ou cartes fusionnés);

De plus, il faut s'assurer que la nouvelle sélection d'éléments ne crée pas de trou dans le réseau routier (il est cependant possible de reclassifier les éléments que l'on veut éliminer afin de préserver la continuité du réseau).

##### 3.1.2 DÉTECTION DES ÉLÉMENTS TROP RAPPROCHÉS

---

La détection des éléments trop rapprochés doit être effectuée dans la mesure du possible après l'édition finale de la carte généralisée.

#### 3.2 PLAN D'ACTION SUGGÉRÉ SUITE AU PRÉSENT PROJET

---

Suite au présent projet, nous suggérons les actions suivantes à l'équipe du Ministère.

- **Expérimentation et exploitation du logiciel**  
Nous suggérons à l'équipe du Ministère de se familiariser avec le logiciel en effectuant la généralisation à partir de données structurées de divers secteurs du Québec.
- **Au besoin, obtenir du support**  
Étant donné que les ressources du Ministère sont assez restreintes, il pourrait être avantageux pour celui-ci d'obtenir du support de la part d'un fournisseur pour améliorer l'environnement de production du Ministère et pour produire les données généralisées. Ces travaux pourraient prendre la forme de prêt de personnel et être effectués dans les bureaux du Ministère.

- **Identification d'améliorations au logiciel**

À la lumière des résultats obtenus suite à l'expérimentation et à l'exploitation du logiciel, le Ministère sera en mesure d'identifier les améliorations les plus prioritaires à apporter au logiciel (voir la section 3.3).

- **Poursuite de l'amélioration du logiciel**

Un autre mandat pourrait être accordé pour l'amélioration du logiciel, en se concentrant sur les améliorations qui rapporteront le maximum de bénéfices au Ministère.

### **3.3 PERSPECTIVES D'AMÉLIORATION FUTURE DU LOGICIEL GÉNÉRALISATION-MTQ**

---

Même à ce stade-ci, plusieurs avenues d'améliorations du logiciel sont envisagées :

- **Développement d'une interface graphique**

Une interface graphique augmenterait très significativement la convivialité du logiciel.

- **Ajout de fonctions de validation visuelle**

Via l'interface graphique, l'utilisateur pourrait comparer la carte originale à la carte généralisée, chacune étant présentée d'une couleur différente. L'utilisateur pourrait reporter des éléments de la carte originale vers la carte généralisée au besoin (dans le cas d'une omission du logiciel) ou éliminer des éléments conservés mais non désirés dans la carte généralisée. Le logiciel de généralisation effectue au moins 90% du travail, mais ces fonctions permettraient de produire un résultat exactement conforme à celui désiré.

- **Ajout de fonctions de déplacement des éléments trop près**

Présentement, l'application ne fait que détecter les éléments trop près. Via l'interface graphique, l'utilisateur pourrait cette fois éliminer ou déplacer certains éléments interactivement.

- **Ajout de fonctions de symbolisation**

En symbolisant à l'écran les éléments tels que représentés sur la carte finale, l'utilisateur pourrait effectuer son travail plus aisément, notamment en ce qui concerne l'édition des éléments trop rapprochés.

- **Lissage adaptif**

Présentement, le lissage des lignes est paramétrisé par une valeur exprimée en pourcentage. Ce pourcentage est appliqué à tous les éléments d'une même classe. En exprimant le degré de lissage par un nombre de vertex désirés par millimètre à l'échelle de la carte, le degré de lissage pourrait s'ajuster plus précisément à chaque élément graphique. Le résultat serait plus homogène.

- **Expérimentation et prise en compte d'autres catégories d'éléments**

Le présent mandat s'est concentré sur le réseau routier et le cadastre. Les travaux pourraient être complétés en ajoutant l'hydrographie, entre autres.

- **Améliorations relatives à la généralisation des zones urbaines**

Des résultats très prometteurs ont déjà été obtenus relativement à la généralisation des zones urbaines. Suite à l'expérimentation par le Ministère, des spécifications plus précises pourront être formulées pour décrire plus exactement les résultats désirés dans les zones urbaines. Par exemple, veut-on des zones tramées ou des vecteurs simplifiés?

Dans le cas où le Ministère désirerait accentuer le travail de généralisation en zone urbaine, des fonctions interactives pourraient être développées. Via une interface graphique, l'utilisateur et le système travailleraient en symbiose. L'utilisateur guiderait le système, qui à son tour, présenterait les résultats des décisions au fur et à mesure. Par exemple, l'utilisateur identifie une artère à éliminer en pointant un de ses segments. Le système détecte les autres segments de cette artère (segments dans le prolongement), et les élimine. L'utilisateur confirme ou annule.

#### 4. CONCLUSIONS

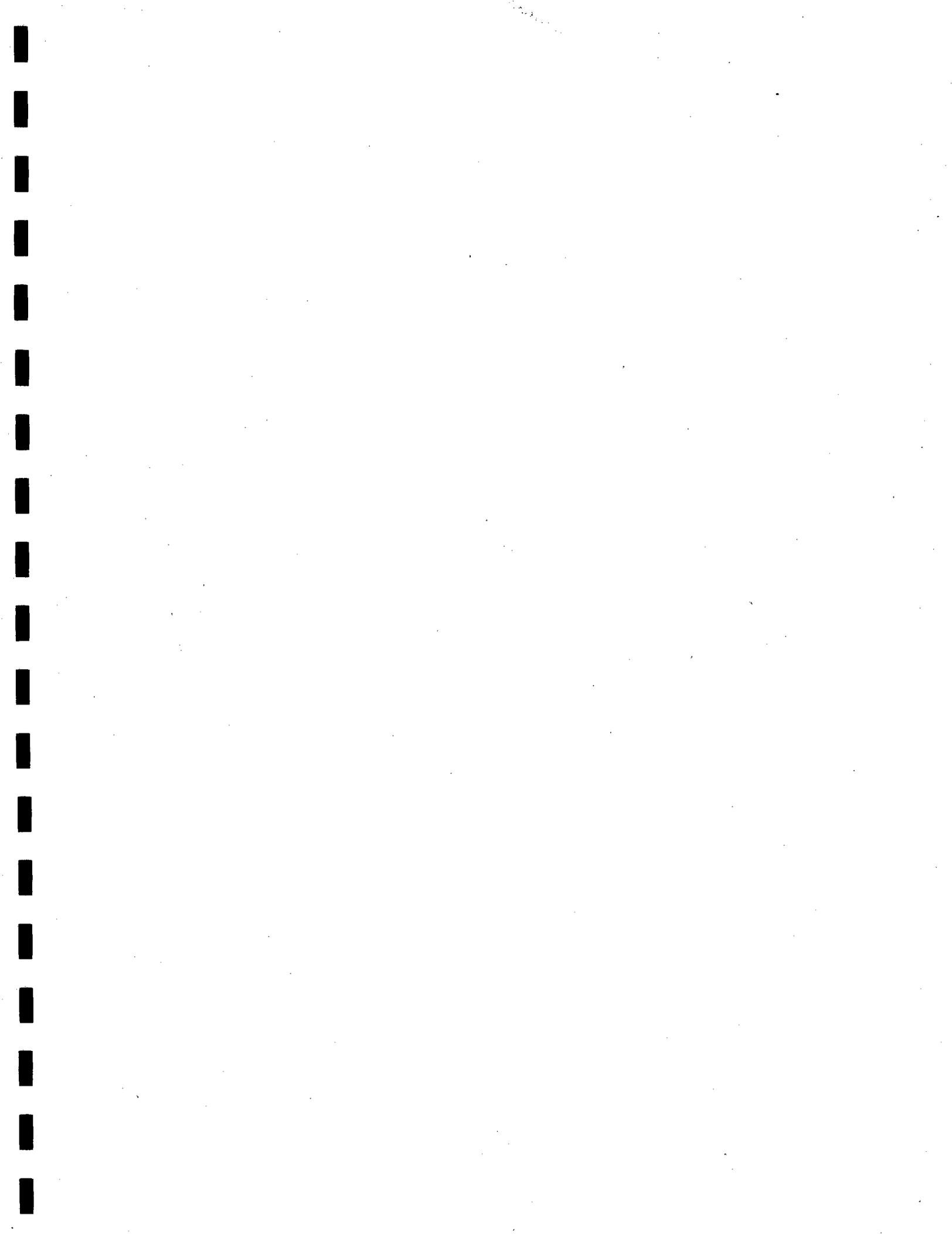
---

Nous sommes convaincus que le présent mandat fut une grande réussite.

Même s'il s'agissait d'un projet de recherche et développement comportant des incertitudes, PGI a réussi à développer des outils pratiques et pouvant rendre des services concrets au Ministère, dans ses travaux de production cartographique.

Ceci est probablement attribuable à l'expérience que PGI possédait dans ce domaine précis, ainsi qu'au calibre des individus ayant oeuvré au projet.

Grâce à ce mandat, le Ministère a fait un progrès très considérable dans le domaine de la généralisation (surtout si l'on considère le budget disponible). Il est fort probable qu'avec les outils développés, le Ministère soit déjà en avance, dans le domaine de la généralisation, sur de nombreuses organisations d'envergure. Nous espérons pouvoir poursuivre ces développements, toujours dans la mesure où ceux-ci peuvent rendre de réels services au Ministère des Transports.



**MINISTÈRE DES TRANSPORTS DU QUÉBEC**  
**SERVICE DE LA CARTOGRAPHIE**

**PROJET DE GÉNÉRALISATION**

**Développement d'outils de généralisation de données cartographiques numériques pour la production de diverses séries cartographiques à petite échelle**

**GUIDE TECHNIQUE**

**PGI** PHOTOSUR GEOMAT  
INTERNATIONAL

**3 JUIN 1997**

TABLE DES MATIÈRES

---

	<u>PAGE</u>
1. OBJET .....	1
2. CONVENTIONS SYNTAXIQUES ET TERMINOLOGIE UTILISÉE .....	2
3. PROCÉDURE D'INSTALLATION DE L'APPLICATION .....	3
3.1 DESCRIPTION DU FICHIER MTQ.TAR.Z .....	3
3.2 EXTRACTION DES FICHIERS .....	3
3.3 STRUCTURE DE RÉPERTOIRE CRÉÉE LORS DE L'EXTRACTION .....	3
3.4 STRUCTURE DE RÉPERTOIRE SUGGÉRÉE .....	4
3.4.1 Structure de répertoire système .....	4
3.4.2 Structure de répertoire projet .....	5
4. PROGRAMMES AML DE L'APPLICATION .....	6
4.1 LISTE DES PROGRAMMES AML .....	6
5. LE PROGRAMME <i>GENLIGNE</i> EN LANGAGE C++ DE L'APPLICATION .....	7
5.1 DESCRIPTION DES FICHIERS INTRANTS DU PROGRAMME <i>GENLIGNE</i> .....	7
5.2 DESCRIPTION DES FICHIERS EXTRANTS DU PROGRAMME <i>GENLIGNE</i> .....	8
6. DIAGRAMMES DES PRINCIPAUX TRAITEMENTS DES AML DE L'APPLICATION .....	10
6.1 DIAGRAMME DES TRAITEMENTS DU PROGRAMME <i>ARC.AML</i> .....	10
6.2 DIAGRAMME DES TRAITEMENTS DU PROGRAMME <i>IMPORTE_DGN.AML</i> .....	11
6.3 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME <i>GEN_MTQ.AML PHASE 1</i> .....	12
6.4 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME <i>GEN_MTQ.AML PHASE 2</i> .....	13
6.5 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME <i>DETEC_LIGNES</i> .....	14

**TABLE DES FIGURES**

---

	<u>PAGE</u>
Figure 1 Structure de répertoire créée lors de l'extraction .....	3
Figure 2 Structure suggérée de répertoire système .....	4
Figure 3 Structure de répertoire minimale suggérée .....	5
Figure 4 Structure de répertoire suggérée pour un environnement particulier .....	5

## 1. OBJET

---

Ce guide s'adresse à l'administrateur de système et a pour but de fournir toutes les indications nécessaires à l'installation et à la configuration de l'application **GÉNÉRALISATION-MTQ**. Des informations concernant le flux de traitement des données sont également incluses dans ce document.

**Important:** L'utilisateur de ce guide doit avoir lu au préalable le Guide de l'utilisateur.

Ce guide fournit:

- la procédure d'installation de l'application;
- la liste des programmes AML;
- la description du programme GENLIGNE en langage C++ ;
- les diagrammes des principaux traitements des programmes AML.

## 2. CONVENTIONS SYNTAXIQUES ET TERMINOLOGIE UTILISÉE

---

Tout au long de ce guide, les conventions syntaxiques présentées ci-dessous sont utilisées pour distinguer les divers éléments de texte.

- gras souligné:** Indique du texte qui doit être saisi au clavier;
- italique gras:*** Indique une commande, un programme ou une directive AML;
- italique:*** Indique du texte qui apparaît à l'écran;
- souligné:** Indique le nom d'un fichier ou d'un répertoire;
- < argument > :** Indique un argument obligatoire d'un programme;
- { argument } :** Indique un argument optionnel d'un programme;
- Police courrier:** Indique le contenu d'un fichier de format ASCII.

### TERMINOLOGIE UTILISÉE:

- Couverture:** Le terme **couverture** représente un ensemble de données vectorielles stockées par le logiciel ARC/INFO.
- Indicatif:** Classe d'élément identifiée par le champ **INDICATIF** d'une **couverture**.

### 3. PROCÉDURE D'INSTALLATION DE L'APPLICATION

---

Ce chapitre décrit la procédure d'installation de l'application.

#### 3.1 DESCRIPTION DU FICHIER MTQ.TAR.Z

---

Le fichier mtq.tar.Z est un fichier de format "TAR" (Unix) comprimé qui contient les programmes, les tables ainsi que des jeux de données selon une structure de répertoire préétablie.

#### 3.2 EXTRACTION DES FICHIERS

---

Les étapes à suivre sont les suivantes:

- copier le fichier mtq.tar.Z dans le répertoire du projet (répertoire vide);  
⇒ Exemple: /usr1/data/projet
- décompresser le fichier à l'aide de la commande suivante:  
⇒ ***uncompress mtq.tar***
- exécuter la commande suivante:  
⇒ ***tar xvf mtq.tar***

L'extraction des fichiers est alors automatiquement exécutée et les sous-répertoires créés dans le répertoire courant.

#### 3.3 STRUCTURE DE RÉPERTOIRE CRÉÉE LORS DE L'EXTRACTION

---

Suite à l'extraction des fichiers, la structure de répertoire suivante est alors créée:

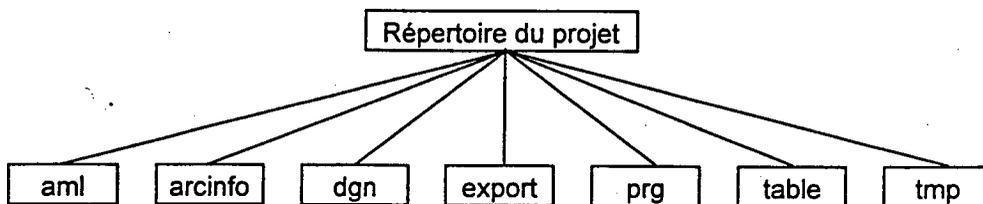


Figure 1 Structure de répertoire créée lors de l'extraction

La description des répertoires est la suivante:

- aml contient tous les programmes sources en langage AML (ARC Macro Language);
- arcinfo contient les couvertures ARC/INFO;
- dgn contient les fichiers DGN;
- export contient les couvertures ARC/INFO en format d'échange EXPORT;
- prg contient le programme exécutable *genligne* en langage C++;
- table contient la table de correspondance des tables de paramètres ainsi que toutes les tables de paramètres.
- tmp contient les fichiers et couvertures temporaires créés par l'application.

Cette structure de répertoire n'est pas obligatoire. Elle présente tous les répertoires dans un même projet. Pour éviter la duplication des programmes ainsi que celle des tables, la section 3.4 décrit une autre alternative.

Les noms des répertoires sont également flexibles. Ils doivent cependant être définis dans le fichier de configuration (voir le Guide de l'utilisateur).

### 3.4 STRUCTURE DE RÉPERTOIRE SUGGÉRÉE

---

Cette section présente les différentes structures de répertoire possibles.

#### 3.4.1 STRUCTURE DE RÉPERTOIRE SYSTÈME

---

La structure de répertoire suggérée pour les programmes ainsi que les tables communes à différents projet est présentée à la figure suivante:

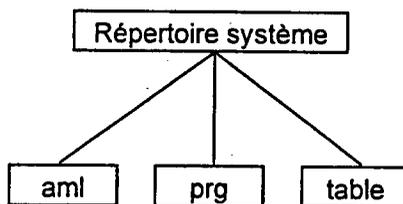


Figure 2 Structure suggérée de répertoire système

En conservant la structure de répertoire système en un seul endroit, les droits d'accès à ces répertoires peuvent facilement être contrôlés. Ceci permet également de modifier les tables en un seul endroit.

### 3.4.2 STRUCTURE DE RÉPERTOIRE PROJET

La structure de répertoire minimale suggérée pour un projet est présentée à la figure suivante:

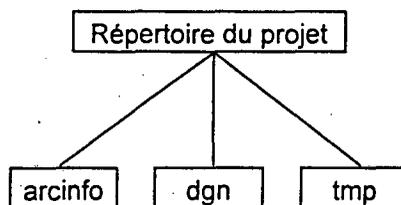


Figure 3 Structure de répertoire minimale suggérée

La structure de répertoire suggérée pour chaque projet nécessitant un environnement particulier est présentée à la figure suivante:

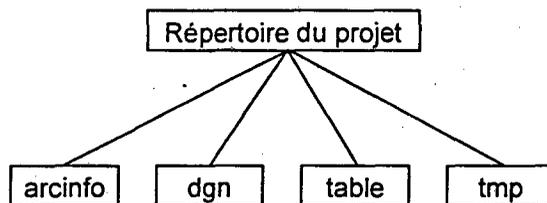


Figure 4 Structure de répertoire suggérée pour un environnement particulier

Les noms de répertoire suggérés aux figures 2, 3 et 4 sont ceux qu'on retrouve dans le fichier de configuration conf\_mtg.ini.

**NOTE:** Il serait possible de définir seulement le répertoire des couvertures ARC/INFO pour un projet et de définir le répertoire des fichiers DGN (mot clé **ReptDgn:**) à l'extérieur du projet. Cette option permet de conserver les DGN comprenant les données sources au même endroit. Ceci s'applique également au répertoire temporaire (mot clé **ReptTmp:**) dans le cas où l'espace disque du répertoire du projet est restreint.

## 4. PROGRAMMES AML DE L'APPLICATION

Ce chapitre décrit les programmes en langage AML de l'application. Lorsqu'il s'agit d'un sous-programme, la chaîne de caractère (s) est ajoutée à la suite du nom du programme et (s-a) lorsque celui-ci peut également fonctionner de façon autonome.

### 4.1 LISTE DES PROGRAMMES AML

Le tableau suivant présente la liste des programmes AML :

Programme	Programme(s) appelant(s)	Programme(s) appelé(s)	Description
<i>arc.aml</i>		<i>gen_init.aml</i>	Initialise une session ARC.
<i>gen_init.aml (s-a)</i>	<i>arc.aml</i> <i>param.aml</i> <i>detec_lignes.aml</i>		Valide le fichier de configuration, l'existence de la table de correspondance et des répertoires associés aux mots clés.
<i>Importe_dgn.aml</i>		<i>stat.aml</i>	Importation de fichier de format DGN.
<i>stat.aml (s-a)</i>	<i>importe_dgn.aml</i>		Génère des statistiques.
<i>gen_mtg.aml</i>		<i>param.aml</i> <i>gen_init.aml</i> <i>change_codes.aml</i>	Programme principal de sélection, filtrage et lissage.
<i>param.aml (s)</i>	<i>gen_mtg.aml</i>	<i>gen_init.aml</i>	Valide la table de paramètres utilisée par <i>gen_mtg.aml</i> .
<i>change_codes.aml (s)</i>	<i>gen_mtg.aml</i>		Change les codes (indicatif) en fonction d'une table de paramètres.
<i>detec_lignes.aml</i>		<i>gen_init.aml</i>	Détection des lignes trop rapprochées.
<i>ae.aml</i>			Démarre un session ARCEDIT et affiche les lignes trop rapprochées.

## 5. LE PROGRAMME GENLIGNE EN LANGAGE C++ DE L'APPLICATION

---

Ce chapitre décrit le programme *genligne* dont la tâche principale consiste à l'élimination des lignes ainsi que le lissage de celles-ci. Ce programme est appelé par le programme principal *gen\_mtq.aml*.

L'exécution de *genligne* sans argument affiche ce qui suit:

*Utilisation:*

*genligne* <Lignes> [<Classes>] <Parametres>

*Les arguments designent des fichiers d'intrants:*

<Lignes>        doit contenir les lignes a generaliser.

<Classes>        contient les classes des lignes selon leur identificateur.  
L'omettre si les identificateurs sont deja des classes.

<Parametres>    doit contenir les parametres de generalisation.

*Les extrants sont places dans des fichiers portant le nom d'un des intrants, mais avec une extension differente:*

<Lignes> .lg     contiendra les lignes generalisees.

<Classes> .tc    contiendra la table des classe des lignes, epuree et trie.

<Lignes> .jrn    contiendra le journal d'execution (erreurs, statistiques...).

L'argument <Lignes>        représente un fichier ASCII généré par la commande *UNGENERATE* du logiciel ARC/INFO.

L'argument <Classes>        représente un fichier ASCII généré par le programme *gen\_mtq.aml*.

L'argument <Parametres>    représente une table de paramètres tel que décrit dans le Guide de l'utilisateur.

### 5.1 DESCRIPTION DES FICHIERS INTRANTS DU PROGRAMME GENLIGNE

---

Les fichiers d'intrants et d'extrants sont localisés dans le répertoire défini par le mot clé **ReptTmp**: et sont automatiquement détruits à la fin de l'exécution du programme *gen\_mtq.aml*.

Le fichier de lignes (argument <Lignes>) contient les lignes à généraliser. Ce fichier est généré par la commande *UNGENERATE* du logiciel ARC/INFO par le biais du programme *gen\_mtq.aml* dont le format est relativement simple. Le nom du fichier porte le nom de l'argument <couverture\_entree> du programme *gen\_mtq.aml* suivi de l'extension .gen. En voici la description:

Id <sub>A</sub>
X <sub>1</sub> Y <sub>1</sub>
X <sub>2</sub> Y <sub>2</sub>

Les lignes sont décrites une à la suite de l'autre dans un ordre quelconque.

```
...  
Xn Yn  
END  
...  
Idz  
X1 Y1  
X2 Y2  
...  
Xn Yn  
END  
END
```

Leur description commence par une ligne contenant le numéro identificateur ( $Id_x$ ) de la ligne. Ce numéro représente le "User-ID" de la ligne. Ce numéro doit être unique pour chacune des lignes. Le "User-ID" est un entier de 0 à 2,147,483,647 qui peut être précédé d'espaces.

Par la suite, une série de lignes énumèrent séquentiellement les coordonnées XY des vertex de la ligne. Les X et Y sont des nombres réels (double précision) précédés d'espaces (facultatif pour le X). Finalement une ligne contenant le mot **END** termine la description de chaque ligne.

À la toute fin, une ligne contenant le mot **END** termine le fichier.

Le fichier des classes de lignes (argument <Classes>) représente un fichier ASCII généré par le programme *gen\_mtg.aml*.

Le fichier des classes de lignes sert à associer les numéros identificateurs des lignes aux noms des classes de lignes. Celles-ci sont identifiées par l'indicatif (chaînes de caractères). En voici la description:

```
Ida'Classe'  
Idb'Classe'  
...  
Idz'Classe'
```

Chaque ligne du fichier associe une ligne, via son identificateur ( $Id_x$ ), à une classe de lignes. L'ordre des associations n'a pas d'importance. Chacune commence par le numéro identificateur de la ligne qui peut être précédé d'espaces. Ce numéro est suivi d'une virgule et du nom de la classe de lignes entre apostrophes.

Il doit y avoir le même nombre d'associations qu'il y a de lignes dans le fichier des lignes. Les lignes associées à une même classe de lignes subiront les mêmes traitements.

Le fichier de paramètres (argument <Paramètres>) représente une table de paramètres tel que décrit dans le Guide de l'utilisateur à la section 7.3.3.

## 5.2 DESCRIPTION DES FICHIERS EXTRANTS DU PROGRAMME GENLIGNE

Le programme *genligne* génère trois fichiers d'extrants:

- le fichier de lignes généralisées  
Il a le même format que le fichier d'intrant et porte également le même nom sauf que l'extension est changée par .lg.
- le fichier de classes des lignes généralisées  
Il a le même format que le fichier d'intrant et porte également le même nom sauf que l'extension est changée par .tc. Il contient les associations des lignes qui ont survécu à la généralisation. Les associations sont triées par identificateur de ligne.

- le fichier journal des lignes traitées

C'est un fichier de texte qui enregistre les événements et les informations produites en cours d'exécution. Il porte le même nom que le fichier de lignes d'intrant sauf que l'extension est changée par .jrn.

L'information primordiale qu'il contient est la réussite ou non de la généralisation. La première ligne contiendra **OK** si la généralisation a réussi. Sinon elle peut contenir **ERREUR** ou **EN COURS** si le programme *genLigne* a rencontré des conditions anormales. Le programme *gen\_mtg.aml* valide la première ligne et affiche le message d'erreur suivant si elle ne contient pas **OK**:

⇒ **ERREUR D'EXECUTION DU PROGRAMME GENLIGNE!!!**

*Voir le fichier journal nom\_du\_fichier\_journal*

## 6. DIAGRAMMES DES PRINCIPAUX TRAITEMENTS DES AML DE L'APPLICATION

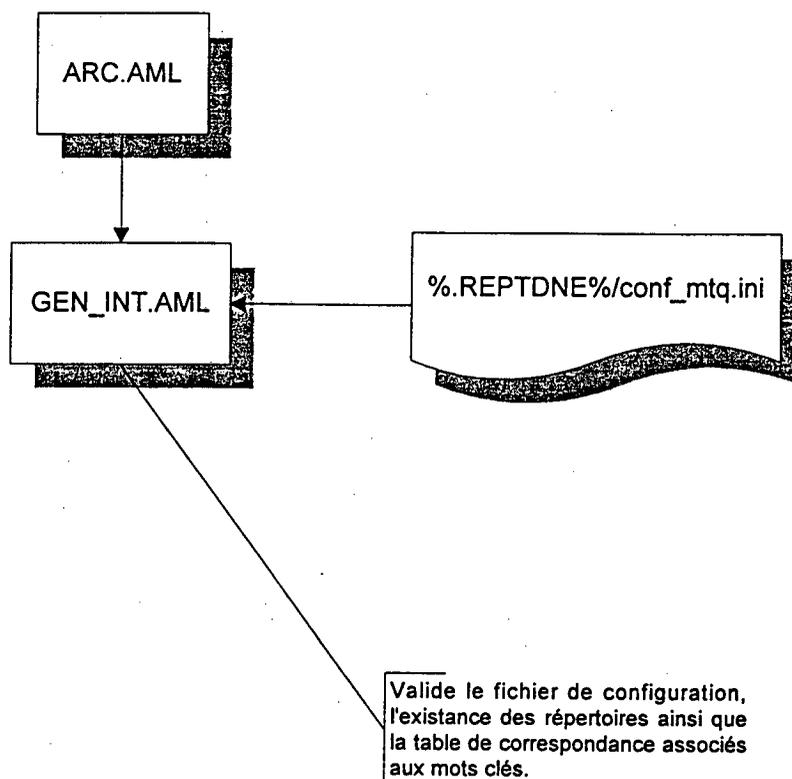
---

Ce chapitre décrit le flux de données des programmes composant l'application GÉNÉRALISATION-MTQ.

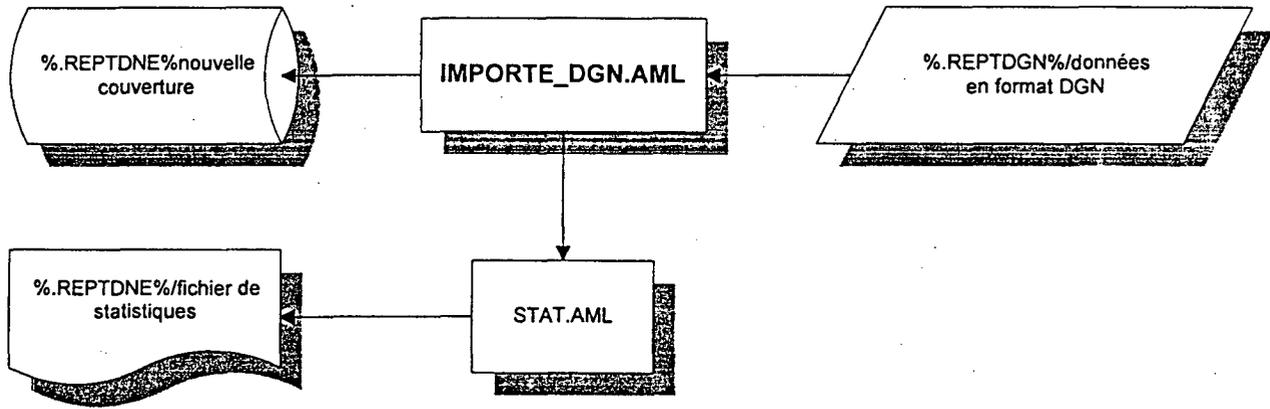
### 6.1 DIAGRAMME DES TRAITEMENTS DU PROGRAMME ARC.AML

---

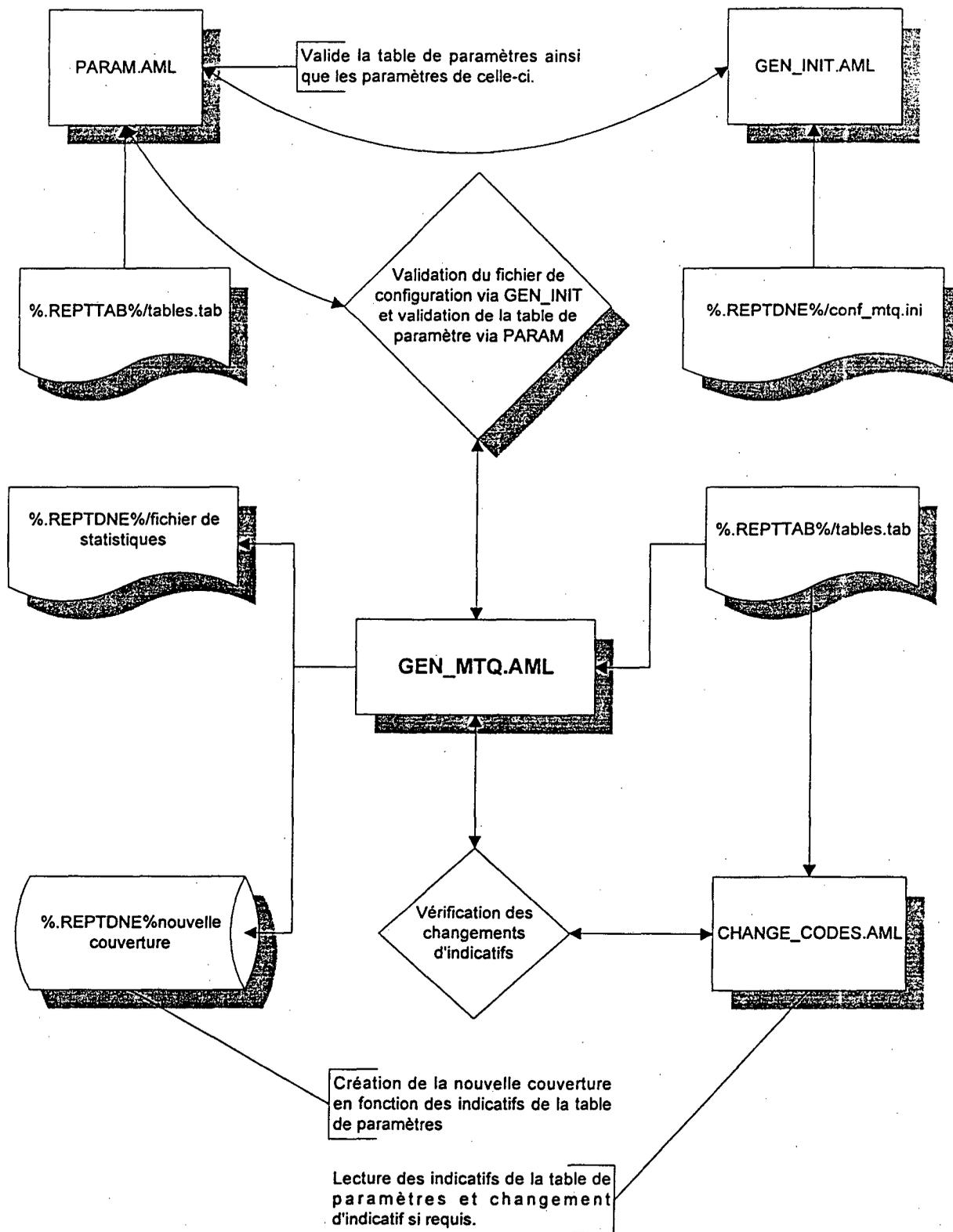
Le programme *arc.aml* doit toujours être utilisé pour débiter une session ARC afin d'initialiser les variables du fichier de configuration. Ce programme doit être localisé dans le répertoire associé au mot clé **ReptDne**: du fichier de configuration. Tous les autres programmes AML doivent être localisés dans le répertoire associé au mot clé **ReptAml**: du fichier de configuration.



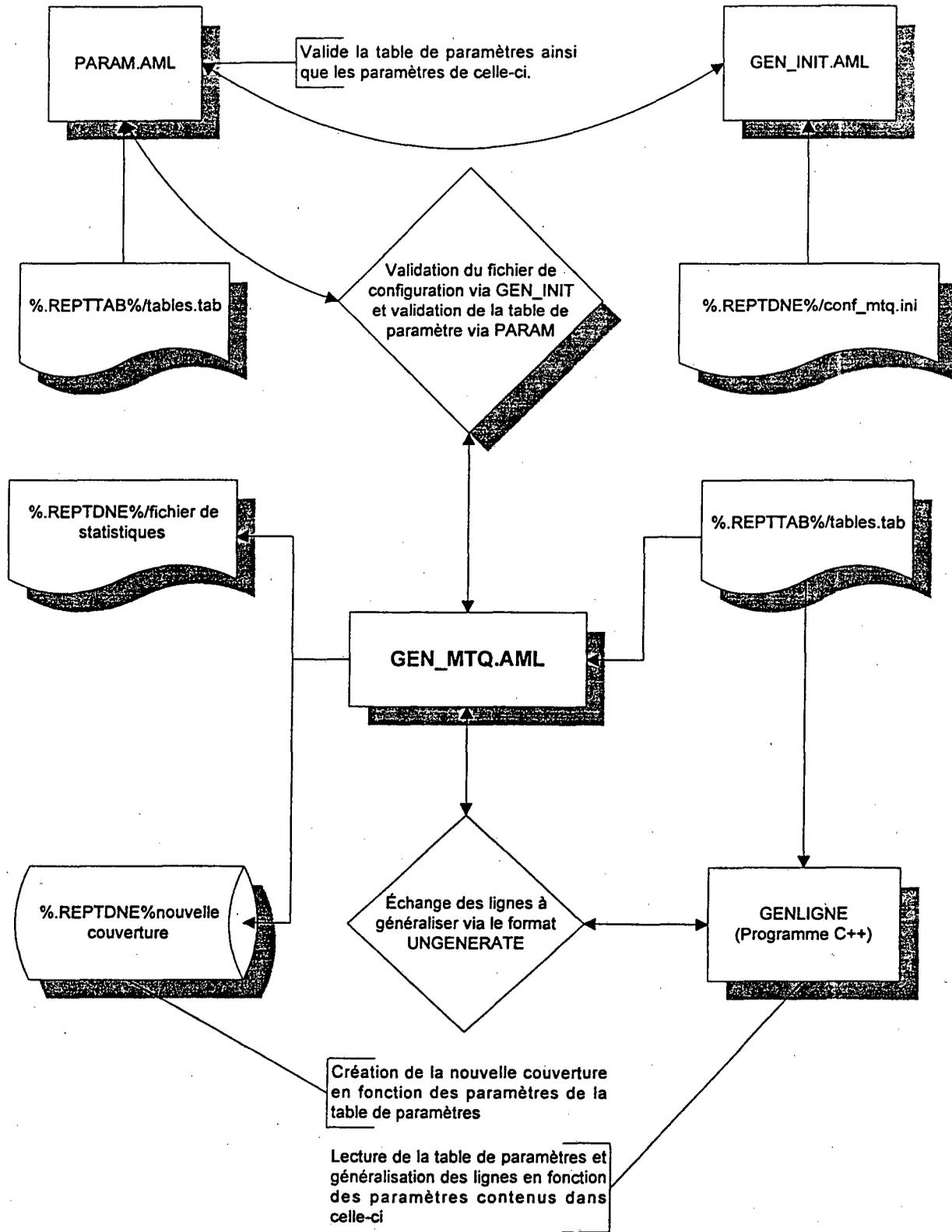
## 6.2 DIAGRAMME DES TRAITEMENTS DU PROGRAMME *IMPORTE\_DGN.AML*



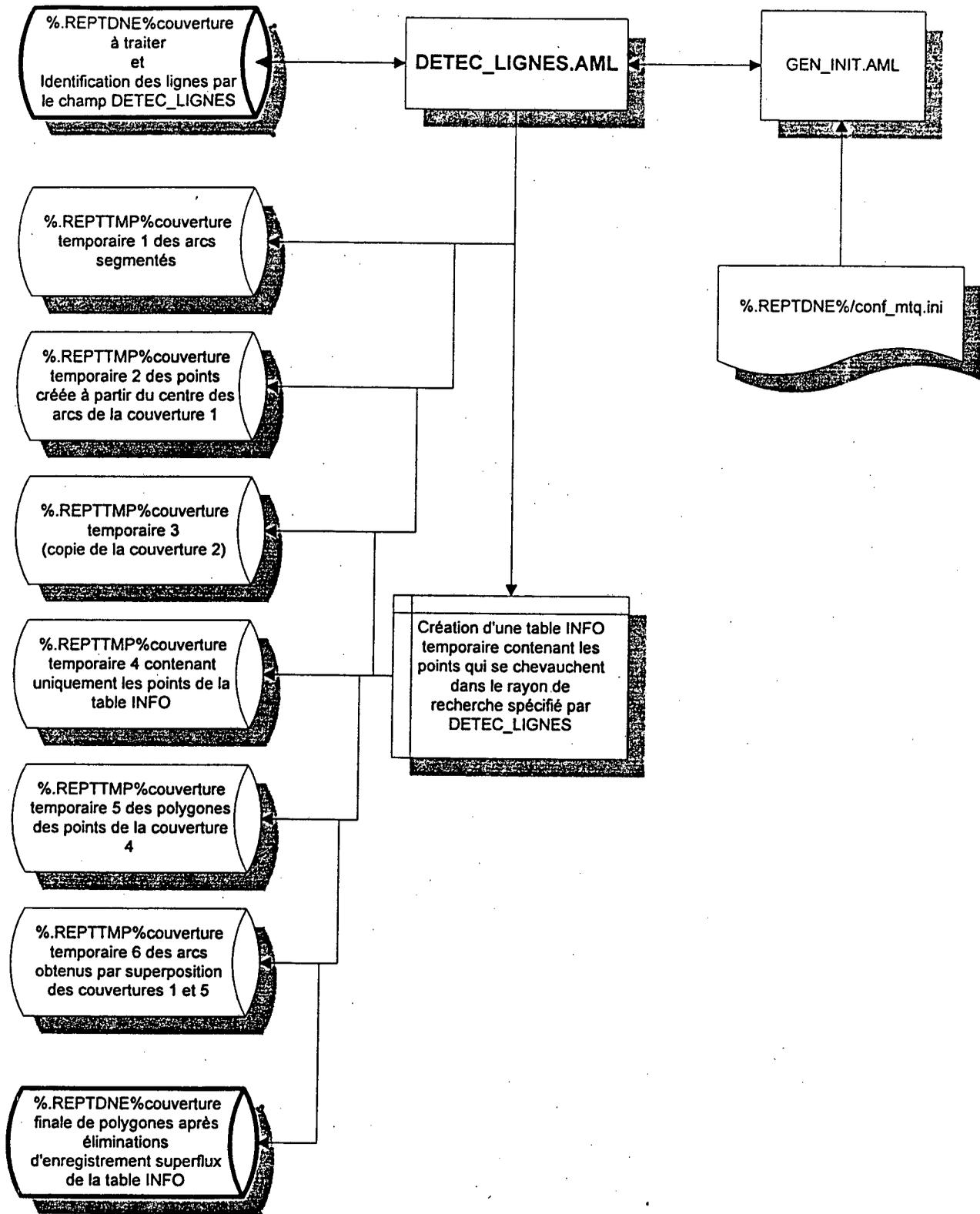
6.3 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME GEN\_MTQ.AML PHASE 1

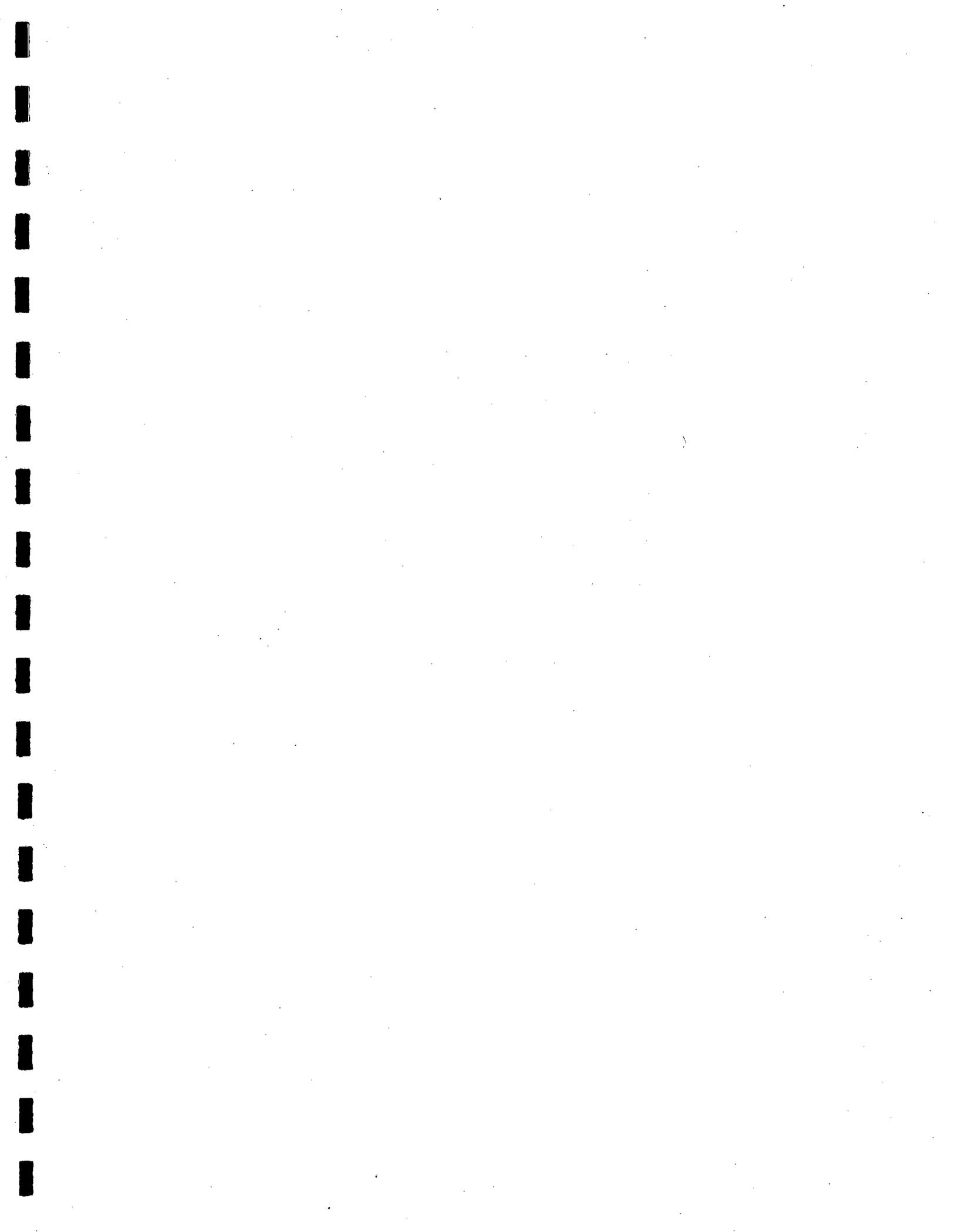


6.4 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME GEN\_MTQ.AML PHASE 2



### 6.5 DIAGRAMME DES PRINCIPAUX TRAITEMENTS DU PROGRAMME DETEC\_LIGNES





MINISTÈRE DES TRANSPORTS DU QUÉBEC  
SERVICE DE LA CARTOGRAPHIE

**PROJET DE GÉNÉRALISATION**

**Développement d'outils de généralisation de données cartographiques numériques pour la production de diverses séries cartographiques à petite échelle**

**GUIDE DE L'USAGER**

**PGI** PHOTOSUR GEOMAT  
INTERNATIONAL

30 MAI 1997

TABLE DES MATIÈRES

	<u>PAGE</u>
1. OBJET .....	1
2. CONVENTIONS SYNTAXIQUES ET TERMINOLOGIE UTILISÉE .....	2
3. DESCRIPTION GÉNÉRALE DE L'APPLICATION .....	3
4. ENVIRONNEMENT DE DÉVELOPPEMENT .....	4
5. PRINCIPES DE BASE DU FONCTIONNEMENT DE L'APPLICATION.....	5
6. LES ÉTAPES DE PRÉPARATION AVANT L'EXPLOITATION DE L'APPLICATION .....	6
6.1 FICHER DE CONFIGURATION .....	6
6.2 TABLE DE CORRESPONDANCE DES TABLES DE PARAMÈTRES.....	7
6.3 STRUCTURE DE RÉPERTOIRE.....	8
6.4 LA COMMANDE <i>ARC.AML</i> DU RÉPERTOIRE DE DONNÉES.....	8
6.5 DÉBUTER UNE SESSION ARC DANS L'ENVIRONNEMENT DE L'APPLICATION.....	9
7. PROGRAMMES COMPOSANT L'APPLICATION .....	10
7.1 PROGRAMME <i>IMPORTE_DGN</i> .....	10
7.1.1 Exemple d'utilisation du programme <i>importe_dgn</i> .....	11
7.2 PROGRAMME <i>STAT</i> .....	12
7.2.1 Exemple d'utilisation du programme <i>stat</i> .....	12
7.3 PROGRAMME <i>GEN_MTQ</i> .....	13
7.3.1 Description de la phase 1 du programme <i>gen_mtg</i> .....	13
7.3.2 Description de la phase 2 du programme <i>gen_mtg</i> .....	14
7.3.3 Description d'une table de paramètres utilisée par le programme <i>gen_mtg</i> .....	14
7.3.3.1 Contenu d'une table de paramètres .....	15
7.3.3.2 Description du format et des paramètres d'une table de paramètres .....	16
7.3.4 Exemple d'utilisation du programme <i>gen_mtg option phase 1</i> .....	23
7.3.5 Exemple d'utilisation du programme <i>gen_mtg option phase 2</i> .....	24
7.4 PROGRAMME <i>DETEC_LIGNES</i> .....	26
7.4.1 Exemple d'utilisation du programme <i>detec_lignes</i> .....	27
7.5 PROGRAMME <i>AE</i> .....	27
7.5.1 Exemple d'utilisation du programme <i>ae</i> .....	28
ANNEXE 1 .....	29
TABLE 20K_50K.TAB .....	29
ANNEXE 2 .....	32
TABLE 20K_125K.TAB .....	32
ANNEXE 3 .....	35
TABLE 20K_500K.TAB .....	35
ANNEXE 4 .....	38
TABLE 20K_1G.TAB .....	38
ANNEXE 5 .....	41
TABLE 50K_125K.TAB .....	41
ANNEXE 6 .....	43
TABLE 50K_500K.TAB .....	43

TABLE DES FIGURES

---

	<u>PAGE</u>
Figure 1 Structure de répertoire minimale suggérée .....	8
Figure 2 Structure de répertoire suggérée pour un environnement particulier .....	8
Figure 3 Exemple de reclassification .....	17
Figure 4 Exemple de pseudo-noeud .....	18
Figure 5 Exemple 1 de Douglas-Peucker .....	19
Figure 6 Exemple 2 de Douglas-Peucker .....	19
Figure 7 Exemple 3 de Douglas-Peucker .....	19
Figure 8 Exemple de 2 noeuds pendants .....	20
Figure 9 Exemple de 1 noeud pendent .....	20
Figure 10 Exemple de 2 noeuds connectés .....	21
Figure 11 Exemple d'un îlot .....	21
Figure 12 Exemple d'un îlot connecté à une autre ligne .....	22
Figure 13 Exemple de détection de lignes trop rapprochées .....	28

## 1. OBJET

---

Ce guide a pour objet d'aider l'utilisateur à maîtriser les principes de base de l'application GÉNÉRALISATION-MTQ. Le guide fournit:

- la description générale de GÉNÉRALISATION-MTQ;
- l'environnement de développement;
- les principes de base de l'application;
- les étapes de préparation avant l'exploitation de l'application;
- la façon d'utiliser l'application;
- les programmes disponibles de l'application;
- la description de chaque programme;
- un exemple d'utilisation pour chaque programme;
- les tables de paramètres de l'application.

## 2. CONVENTIONS SYNTAXIQUES ET TERMINOLOGIE UTILISÉE

---

Tout au long de ce guide, les conventions syntaxiques présentées ci-dessous sont utilisées pour distinguer les divers éléments de texte.

- gras souligné:** Indique du texte qui doit être saisi au clavier;
- italique gras:*** Indique une commande, un programme ou une directive AML;
- italique:*** Indique du texte qui apparaît à l'écran;
- souligné:** Indique le nom d'un fichier ou d'un répertoire;
- < argument > :** Indique un argument obligatoire d'un programme;
- { argument } :** Indique un argument optionnel d'un programme;
- Police courrier:** Indique le contenu d'un fichier de format ASCII.

### TERMINOLOGIE

#### UTILISÉE:

---

- Couverture:** Le terme **couverture** représente un ensemble de données vectorielles stockées par le logiciel ARC/INFO.
- Indicatif:** Classe d'élément identifiée par le champ **INDICATIF** d'une **couverture**.

### 3. DESCRIPTION GÉNÉRALE DE L'APPLICATION

---

**GÉNÉRALISATION-MTQ** est une application ARC/INFO qui permet l'automatisation d'une partie du processus de généralisation de données cartographiques. L'application a été développée en fonction des besoins du Service de la cartographie du Ministère des Transports du Québec.

L'application s'appuie sur le mode de traitement en lot. La configuration et les paramètres sont contrôlés par le biais de tables de format ASCII.

L'application permet les traitements suivants:

- importation de fichier de format DGN standard ou modifié pour ARC/INFO;
- statistiques des lignes et textes des **couvertures** ARC/INFO;
- création de nouvelles **couvertures** (données de sortie) en fonction de la sélection d'entités de la **couverture** originale (**couverture d'entrée**) via une table de paramètres;
- sélection du type d'entité (ligne, texte et point) requis dans la nouvelle **couverture**;
- changement de l'indicatif des entités lors de la création de la nouvelle **couverture**;
- reclassification des lignes en fonction d'un critère de longueur;
- élimination des pseudo-noeuds;
- élagage des lignes selon l'algorithme de Douglas-Peucker;
- élimination de lignes comportant 2 noeuds pendants selon un critère de longueur;
- élimination de lignes comportant 1 noeud pendant selon un critère de longueur;
- élimination de lignes comportant 2 noeuds connectés selon un critère de longueur et un angle de déflexion;
- élimination de lignes formant un îlot selon un critère de longueur;
- élimination de lignes formant un îlot connecté à une ou plusieurs lignes selon un critère de longueur;
- lissage des lignes en fonction d'un pourcentage de points à conserver;
- création de noeuds virtuels lors du traitement de lissage afin de conserver certains vertex;
- détection des lignes trop rapprochées selon un rayon de recherche.

De plus, chaque traitement peut être paramétrisé de façon indépendante pour chacune des classes d'éléments.

#### 4. ENVIRONNEMENT DE DÉVELOPPEMENT

---

L'application **GÉNÉRALISATION-MTQ** a été développée dans l'environnement de programmation AML du logiciel ARC/INFO version 7 (Unix). Le programme principal (AML) fait appel à plusieurs autres AML ainsi qu'un programme en langage C++ dont la tâche principale consiste à l'élimination des lignes ainsi que le lissage de celles-ci. Le format d'échange entre la couverture et le programme C++ est le format ASCII créé par la commande **UNGENERATE** du logiciel ARC/INFO.

## 5. PRINCIPES DE BASE DU FONCTIONNEMENT DE L'APPLICATION

---

La configuration et la structure de répertoire doivent être définies selon le chapitre 6.

Les couvertures ARC/INFO qui peuvent être utilisées sont les suivantes:

- via le programme d'importation de fichier de format DGN. Deux types de format DGN sont acceptés:
  - ⇒ Standard comportant l'indicatif ("Graphic Group") tel que le devis C du MRN;
  - ⇒ DGN de type devis D du MRN modifié avec un fichier d'attribut ATT fourni par le Service de la Cartographie du MRN.
- toutes autres couvertures ARC/INFO qui comportent un champ selon la structure suivante:

ITEM NAME	WIDTH	OUTPUT	TYPE
INDICATIF	11	11	C

La plupart des programmes de GÉNÉRALISATION-MTQ utilisent une table de paramètres de format ASCII. Une ligne dont le premier caractère est un # est considérée comme un commentaire. Les paramètres représentant une longueur ou une distance sont exprimés en millimètres à l'échelle de la carte.

La première étape (phase 1) consiste à la création d'une nouvelle couverture selon les indicatifs requis par le MTQ en fonction des échelles de sortie. Par la suite, on doit s'assurer que la structure des données respecte les points suivants:

- les noeuds doivent être existants à tous les points d'intersection;
- le réseau routier doit être continu.

De plus, il faut s'assurer que la nouvelle sélection d'éléments ne crée pas de trou dans le réseau routier (il est cependant possible de reclassifier les éléments que l'on veut éliminer afin de préserver la continuité du réseau).

La deuxième étape (phase 2) consiste à l'élimination des lignes (filtrage) ainsi que le lissage de celles-ci.

Par la suite, il est possible de procéder à la détection des lignes trop rapprochées. Le programme identifie les lignes par le biais d'un champ spécifique. Lorsque des lignes sont identifiées, une couverture de polygone est alors créée afin de mieux identifier le ou les segments de la ligne détectée. Une commande AML permet de visualiser la couverture comportant des lignes identifiées comme étant trop rapprochées. Les lignes détectées sont alors affichées en rouge, tandis que la couverture de polygones est affichée en vert.

## 6. LES ÉTAPES DE PRÉPARATION AVANT L'EXPLOITATION DE L'APPLICATION

---

Ce chapitre décrit les principales étapes à suivre avant l'exploitation de l'application.

### 6.1 FICHER DE CONFIGURATION

---

Le fichier de configuration permet d'identifier la localisation de chacun des répertoires requis pour l'application ainsi que la localisation du fichier de correspondance des tables.

Voici un exemple du contenu du fichier de configuration:

```
#=====
# fichier de configuration GENERALISATION-MTQ
# Nom du fichier: conf_mtg.ini
#=====

ReptProj: /data/louis/mtq # Repertoire du Projet
ReptAml: %.ReptProj%/aml # Repertoire des aml
ReptDne: %.ReptProj%/arcinfo # repertoire des donnees ARC/INFO
ReptDgn: %.ReptProj%/dgn # repertoire des fichiers DGN
ReptTab: %.ReptProj%/table # Repertoire des tables
ReptPrg: %.ReptProj%/prg # Repertoire des programmes executables
ReptTmp: %.ReptProj%/tmp # Repertoire des fichiers temporaires

TabTab: tables.tab # Table de correspondance des tables de parametres
```

Une ligne dont le premier caractère est un # est considérée comme un commentaire. Chaque ligne débute avec un mot clé. Celui-ci est par la suite associé à une variable globale dans ARC/INFO. Voici une brève description de chacun des mots clés:

- ReptProj: répertoire principal du projet. Il est possible d'utiliser la variable associée à ce mot clé pour identifier les répertoires parents du projet. Il suffit d'utiliser la syntaxe suivante pour identifier cette variable: %.ReptProj% ;
- ReptAml: répertoire contenant les AML de l'application;
- ReptDne: répertoire des données ARC/INFO;
- ReptDgn: répertoire des fichiers DGN;
- ReptTab: répertoire des tables de paramètres;

- ReptPrg: répertoire du programme exécutable C++
- ReptTmp: répertoire des fichiers temporaires créés par l'application;
- TabTab: fichier de correspondance des tables de paramètres.

Le fichier de configuration conf\_mtg.ini fourni avec l'application peut être utilisé comme modèle et édité à l'aide d'un éditeur de texte afin d'identifier le nouveau projet.

**Important:** Le fichier de configuration doit être localisé dans le répertoire de données ARC/INFO ("Workspace") associé au mot clé ReptDne.

## 6.2 TABLE DE CORRESPONDANCE DES TABLES DE PARAMÈTRES

---

Le fichier défini par le mot clé "TabTab:" contient les valeurs associées aux échelles d'entrée et sortie ainsi que les noms des tables correspondantes. Il est lu et validé lors de l'exécution du programme principal gen\_mtg. Il doit également être localisé dans le répertoire des tables de paramètres.

Voici un exemple du contenu du fichier tables.tab fourni avec l'application:

```
#=====
# Table de correspondance des tables de parametres associes
# aux echelles entree -> sortie
# Nom du fichier: tables.tab
#=====
# Echelle      Echelle      Nom de la
# entree      sortie      table
#=====
20k           50k          20k_50k.tab
20k           125k         20k_125k.tab
20k           500k         20k_500k.tab
20k           1g           20k_1g.tab
50k           125k         50k_125k.tab
50k           500k         50k_500k.tab
```

**Important:** Les valeurs associées aux échelles d'entrée et sortie sont les mêmes qui doivent être utilisées lors de l'exécution du programme principal *gen\_mtg*.

### 6.3 STRUCTURE DE RÉPERTOIRE

---

La structure de répertoire minimale suggérée pour un projet est présentée dans la figure suivante:

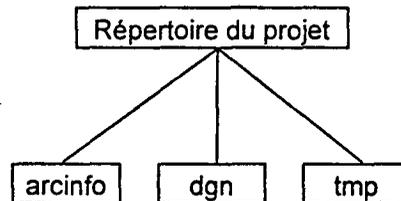


Figure 1 Structure de répertoire minimale suggérée

La structure de répertoire suggérée pour chaque projet nécessitant un environnement particulier est présentée dans la figure suivante:

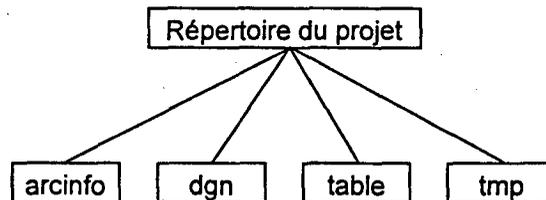


Figure 2 Structure de répertoire suggérée pour un environnement particulier

Les figures ci-dessus présentent les noms de répertoire du fichier *conf\_mtg.ini*

### 6.4 LA COMMANDE *ARC.AML* DU RÉPERTOIRE DE DONNÉES

---

Avant de lancer la commande *arc* pour débiter une session ARC, on doit éditer le fichier de commande *arc.aml* localisé dans le répertoire de données.

Voici un exemple du contenu du fichier de commande *arc.aml* :

```
&amlpath [JOINFILE /usr1/data/projet aml -sub]
&r gen_init conf_mtg.ini
```

Il suffit de changer le chemin d'accès /usr1/data/projet par le répertoire principal contenant le répertoire des AML de l'application.

## 6.5 DÉBUTER UNE SESSION ARC DANS L'ENVIRONNEMENT DE L'APPLICATION

---

Pour débiter une session ARC, il suffit de taper la commande *arc*.

Le programme *gen\_init* est alors exécuté et les variables associées aux mots clés du fichier de configuration sont assignées. Les messages du logiciel "ARC/INFO" sont affichés ainsi que les messages suivants:

*Initialisation des parametres en cours...*

*Lecture du fichier de configuration en cours...*

*Les variables sont assignees.*

*Arc:*

Si le répertoire de données ne comporte pas de sous-répertoire INFO, un message est affiché à cet effet. Il suffit d'utiliser la commande *createworkspace* selon la syntaxe suivante:

⇒ *createworkspace* .

Par la suite, quittez la session ARC, et redémarrez de nouveau une session ARC.

## 7. PROGRAMMES COMPOSANT L'APPLICATION

---

Ce chapitre décrit les principaux programmes disponibles dans l'application. Afin d'afficher les arguments requis pour un programme, il suffit de lancer le programme à l'aide de la directive AML *&run* ou *&r* selon la syntaxe suivante:

⇒ *&r nom\_du\_programme*

La liste des programmes disponibles est la suivante:

- *importe\_dgn* (importation de fichier de format DGN)
- *stat* (production des statistiques)
- *gen\_mtq* (programme principal de sélection, filtrage et lissage)
- *detec\_lignes* (détection des lignes trop rapprochées)
- *ae* (démarré une session ARCEDIT et affiche les lignes trop rapprochées)

### 7.1 PROGRAMME *IMPORTE\_DGN*

---

Ce programme permet d'importer des fichiers de format DGN de deux types différents. L'exemple suivant présente les arguments de la commande lorsque celle-ci est exécutée sans aucun argument:

*Arc: &r importe\_dgn*

*Usage: importe\_dgn <type de DGN> <fichier DGN> <couverture de sortie>  
{fichier ATT}*

*Type de DGN = att (DGN devis D(MRN) modifié pour ARC/INFO avec fichier .att)  
= st (DGN comportant l'indicatif (Graphic Group))  
(DGN produit selon le devis C (MRN))*

*L'argument {fichier ATT} est optionnel; il n'est requis qu'avec le type att*

Deux types sont possibles pour l'argument *<type de DGN>*:

- ⇒ **att** pour les fichiers DGN fournis par le MRN selon le devis D et modifié pour ARC/INFO; Ces fichiers portent habituellement l'extension *.swa* et un fichier d'attribut *.att* est également fourni.
- ⇒ **st** pour les fichiers DGN standards comportant l'indicatif ("Graphic Group") tels que les fichiers du devis C du MRN.

L'argument *{fichier ATT}* est optionnel; il n'est requis qu'avec le type **att** afin d'identifier le fichier d'attribut associé au DGN.

Les fichiers DGN et ATT doivent être localisés dans le répertoire des DGN.

La couverture de sortie est localisée dans le répertoire des données ARC/INFO.

**7.1.1 EXEMPLE D'UTILISATION DU PROGRAMME *IMPORTE\_DGN***

---

*Arc: &r importe\_dgn st 31he101c.dgn he101c*

*Debut des traitements a 10.00.00 le 10/05/1997*

*Importation du fichier "31he101c.dgn" en cours...*

- Plusieurs messages sont alors affichés par la commande IGDSARC de ARC/INFO ainsi que par le module INFO.

...

*Creation du fichier de statistiques final des lignes de "he101c" en cours...*

*Creation du fichier de statistiques final des textes de "he101c" en cours...*

*Statistiques des lignes pour la couverture "he101c"*

<i>INDICATIF</i>	<i>OCCURRENCES</i>
<i>23000</i>	<i>28</i>
<i>23020</i>	<i>61</i>

...

*Statistiques des textes pour la couverture "he101c"*

<i>INDICATIF</i>	<i>OCCURRENCES</i>
<i>23200</i>	<i>37</i>
<i>23220</i>	<i>51</i>

...

*Fin des traitements a 10.01.00 le 10/05/1997*

*Arc:*

Un fichier contenant les statistiques affichées à l'écran est automatiquement créé et porte le nom de la **couverture** de sortie suivi de l'extension **.stat**. Il est localisé dans le répertoire des données ARC/INFO.

## 7.2 PROGRAMME STAT

---

Ce programme permet d'afficher et de générer un fichier de statistiques pour une **couverture** ARC/INFO. L'exemple suivant présente les arguments de la commande lorsque celle-ci est exécutée sans aucun argument:

*Arc: &r stat*

*Usage: stat <couverture d'entree> <fichier de statistiques>*

L'argument <couverture d'entree> représente une **couverture** ARC/INFO.

L'argument <fichier de statistiques> représente le nom du fichier de statistiques. Celui-ci est localisé dans le répertoire des données ARC/INFO.

**NOTE:** Il est à noter que la plupart des programmes produisent des statistiques.

### 7.2.1 EXEMPLE D'UTILISATION DU PROGRAMME STAT

---

*Arc: &r stat he101c he101c.stat*

*Creation des statistiques de "he101c" en cours...*

*Creation du fichier de statistiques final des lignes de "he101c" en cours...*

*Creation du fichier de statistiques final des textes de "he101c" en cours...*

*Statistiques des lignes pour la couverture "he101c"*

<i>INDICATIF</i>	<i>OCCURRENCES</i>
------------------	--------------------

<i>23000</i>	<i>28</i>
--------------	-----------

<i>23020</i>	<i>61</i>
--------------	-----------

*...*

*Statistiques des textes pour la couverture "he101c"*

<i>INDICATIF</i>	<i>OCCURRENCES</i>
------------------	--------------------

<i>23200</i>	<i>37</i>
--------------	-----------

<i>23220</i>	<i>51</i>
--------------	-----------

*...*

*Arc:*

### 7.3 PROGRAMME GEN\_MTQ

---

Ce programme permet de générer une nouvelle couverture ARC/INFO selon le choix de la phase1 ou de la phase2. L'exemple suivant présente les arguments de la commande lorsque celle-ci est exécutée sans aucun argument:

*Arc: &r gen\_mtg*

*Usage: gen\_mtg <echelle\_entree> <echelle\_sortie>  
<couverture\_entree> <couverture\_sortie> <phase=>*

*Exemple: gen\_mtg 20k 50k he201 he201\_mtg phase=2*

*Les phases valides sont les suivantes:*

*phase 1: Selection des elements selon les echelles d'entree et sortie*

*phase 2: Generalisation et lissage des elements selon les echelles d'entree et sortie*

L'argument <echelle\_entree> représente la valeur de l'échelle de la **couverture** d'entrée.

L'argument <echelle\_sortie> représente la valeur de l'échelle de la **couverture** de sortie.

Ces valeurs sont associées à la table de correspondance des tables de paramètres.

L'argument <couverture\_entree> représente la **couverture** d'entrée à traiter.

L'argument <couverture\_sortie> représente la nouvelle **couverture** à générer.

**NOTE:** Il est à noter que si l'argument <couverture\_sortie> représente une **couverture** existante, celle-ci est détruite lors du traitement.

L'argument <phase=> représente la phase de traitement que l'on désire effectuer. Il est possible d'entrer cet argument en utilisant **ph=1** ou **ph=2**.

#### 7.3.1 DESCRIPTION DE LA PHASE 1 DU PROGRAMME GEN\_MTQ

---

La phase 1 permet de générer une nouvelle **couverture** en transférant les entités sélectionnées à l'aide des indicatifs contenus dans une table de paramètres. Celle-ci est associée aux arguments des échelles d'entrée et sortie. Ces arguments doivent être présents dans la table de correspondance des tables de paramètres (mot clé du fichier de configuration = TabTab:). La table de paramètres contient tous les **indicatifs** des entités que l'on désire transférer dans la **couverture** de sortie. Il est également possible de changer l'**indicatif** des entités à la sortie ainsi que de reclassifier certaines entités linéaires en fonction d'un critère de longueur. (Voir la description d'une table de paramètres décrite à la section 7.3.3)

L'argument <couverture\_entree> représente la **couverture** d'entrée à traiter. Celle-ci peut contenir des entités de type linéaire, ponctuel ou textuel.

L'argument <couverture\_sortie> représente la nouvelle **couverture** à générer contenant les entités sélectionnées. La **couverture** des entités linéaires portera le nom associé à l'argument, sans extension. Si une **couverture** d'entités textuelles est créée, celle-ci portera le nom de l'argument suivi des caractères **\_T** alors que celle des entités ponctuelles sera suivi des caractères **\_P**.

### 7.3.2 DESCRIPTION DE LA PHASE 2 DU PROGRAMME *GEN\_MTQ*

La phase 2 génère une nouvelle **couverture** et utilise également une table de paramètres. Le traitement associé à cette phase comprend l'élimination de lignes (filtrage) ainsi que le lissage de celles-ci. Le programme principal *gen\_mtg* fait appel au programme *genligne* en langage C++ pour effectuer ces traitements. Les traitements effectués lors de la phase 2 sont les suivants:

- élimination des pseudo-noeuds;
- élagage des lignes selon l'algorithme de Douglas-Peucker;
- élimination de lignes comportant 2 noeuds pendants selon un critère de longueur;
- élimination de lignes comportant 1 noeud pendant selon un critère de longueur;
- élimination de lignes comportant 2 noeuds connectés selon un critère de longueur et un angle de déflexion;
- élimination de lignes formant un îlot selon un critère de longueur;
- élimination de lignes formant un îlot connecté à une ou plusieurs lignes selon un critère de longueur;
- lissage des lignes en fonction d'un pourcentage de points à conserver;
- création de noeuds virtuels lors du traitement de lissage afin de conserver certains vertex.

La description détaillée de ces traitements est décrite à la section 7.3.3.

L'argument <couverture\_entree> représente la **couverture** d'entrée à traiter. Seules les entités de type linéaire sont traitées.

L'argument <couverture\_sortie> représente la nouvelle **couverture** à générer. Celle-ci représente la **couverture** d'entrée généralisée en fonction des paramètres de la table de paramètres.

### 7.3.3 DESCRIPTION D'UNE TABLE DE PARAMÈTRES UTILISÉE PAR LE PROGRAMME *GEN\_MTQ*

Cette table contient tous les paramètres requis lors de l'exécution du programme *gen\_mtg*.

### 7.3.3.1 Contenu d'une table de paramètres

L'exemple suivant présente une partie du contenu du fichier 20k\_50k.tab fourni avec l'application. Le contenu intégral de cette table est fourni à l'annexe 1.

```
# Table 20k 50k
#=====
# Echelle_entree Echelle_sortie
Echelles= 20000 50000
# Unite de resolution
Resolution= M
#=====
#TYPE INDICATIF  INDICATIF PN RECL DP  2NP 1NP  2NC  A_2NC ILOT ILOT_C LIS  N_V  L_V  A_V  DI_V
#   ENTREE      SORTIE
#=====
L  02010000006 02010000006 o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02010001001 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02010001002 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02010025011 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o  10.0 170.0  10.0
L  02010025015 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o  10.0 170.0  10.0
L  02010025025 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o  10.0 170.0  10.0
L  02010025031 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   8.0 170.0   8.0
L  02010025035 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   8.0 170.0   8.0
L  02010025041 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02010025045 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02010003011 @          o  0.0 0.1  7.0  3.0 10.0 25.0  20.0 10.0  0.8  o   2.0 140.0   2.0
L  02010003015 @          o  0.0 0.1  7.0  3.0 10.0 25.0  20.0 10.0  0.8  o   2.0 140.0   2.0
L  02010003021 @          o  0.0 0.1  7.0  3.0 10.0 25.0  20.0 10.0  0.8  o   0.1 160.0   0.1
L  02010010000 @          o  0.0 0.1 10.0  8.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
L  02600030002 @          o  0.0 0.1  7.0  3.0  5.0 25.0  20.0 10.0  0.8  o   5.0 120.0  5.0
# Donnees cadastrales
L  23020      @          o  0.0 0.0  0.0  0.0  0.0 25.0  0.0 10.0  1.0  o   5.0 120.0  5.0
L  23021      @          o  0.0 0.0  0.0  0.0  0.0 25.0  0.0 10.0  1.0  o   5.0 120.0  5.0
T  23220      @          o  0.0 0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.0  o   5.0 120.0  5.0
```

### 7.3.3.2 Description du format et des paramètres d'une table de paramètres

---

Cette section est divisée en trois parties:

- description générale de la table;
- les options utilisées par le programme principal *gen\_mtg* (AML) lors de la phase 1;
- les options utilisées par le programme *genligne* (C++) lors de la phase 2.

#### 1 - Description générale de la table.

Cette table est de format ASCII. Elle peut donc être éditée à l'aide de n'importe quel éditeur de texte.

- Le caractère # permet de mettre des lignes en commentaire. Ces lignes sont alors ignorées.
- Le caractère d'espace est utilisé comme séparateur de champ.
- La première ligne qui ne débute pas par le caractère # doit contenir la chaîne de caractère **Echelles=** suivie d'un caractère d'espace et des valeurs associées aux échelles d'entrée et sortie.  
Exemple: **Echelles= 20000 50000**  
Ces valeurs doivent être exprimées en nombres entiers.
- La ligne suivante doit contenir la chaîne de caractère **Resolution=** suivie d'un caractère d'espace et de l'unité de résolution. Les valeurs possibles sont les suivantes:

**M** (mètre)  
**DM** (décimètre)  
**CM** (centimètre)  
**MM** (millimètre)

Exemple: **Resolution= M**

- ⇒ **NOTE1:** Les valeurs associées à une longueur sont toujours exprimées en millimètres à l'échelle de la carte (nombre réel, ex.: 1.0).
- ⇒ **NOTE2:** Les valeurs associées à un angle sont toujours exprimées en degrés décimaux (nombre réel, ex.: 1.0).

2 - Les options utilisées par le programme principal *gen\_mtq* (AML) lors de la phase 1.

- **TYPE**

Types d'entités à transférer dans la couverture de sortie. Les types valides sont les suivants: L pour entité linéaire, T pour textuelle, P pour ponctuelle ou les combinaisons suivantes: \* pour toutes, LP, LT et PT.

- **INDICATIF ENTREE**

Indicatif ("Graphic Group"), champ alphanumérique, maximum de 11 caractères.

- **INDICATIF SORTIE**

Indicatif ("Graphic Group"), champ alphanumérique, maximum de 11 caractères. Si l'indicatif de sortie est égal à l'indicatif d'entrée, il est possible d'utiliser le caractère @ pour l'indiquer.

- **PN** (Voir la phase 2)

- **RECL**

Longueur maximale des éléments à reclassifier. Toutes les lignes plus courtes que la longueur spécifiée seront reclassifiées. Le caractère "-" signifie que l'élément doit être reclassifié sans aucun critère de longueur. La valeur 0.0 permet de ne reclassifier aucun élément. Les éléments reclassifiés prennent l'indicatif d'un des éléments adjacents.

Exemple: Les ponts sont reclassifiés selon l'indicatif d'une des routes connectée au pont.

Routes de catégorie "classe A"

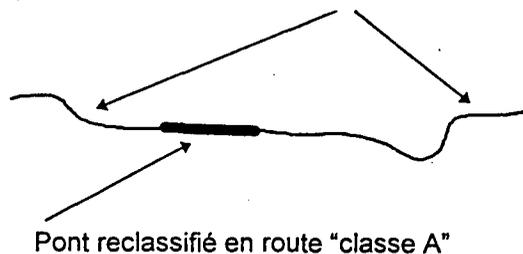


Figure 3 Exemple de reclassification

### 3 - Les options utilisées par le programme *genligne* (C++) lors de la phase 2.

- **PN**

Indiquer par **O** (oui) ou **N** (non) si les pseudo-noeuds doivent être enlevés. Un pseudo-noeud représente la jonction de deux lignes connectées ayant la même classe. En éliminant le pseudo-noeud, les deux lignes seront fusionnées en une seule ligne.

Exemple: Lignes classe A

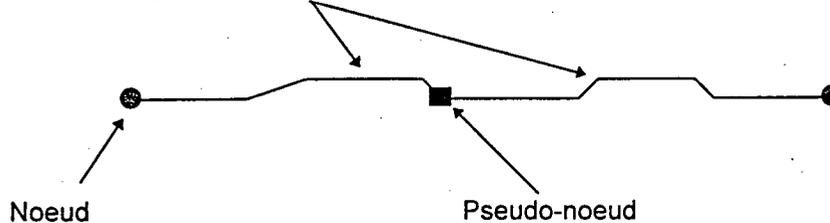


Figure 4 Exemple de pseudo-noeud

⇒ **NOTE:** Il est à noter que la majorité des pseudo-noeuds sont éliminés de façon systématique lors de la phase 1. Cette option permet d'éliminer en plus les pseudo-noeuds créés par ARC/INFO pour les lignes composées de plus de 500 vertex (contrainte ARC/INFO).

### OPTIONS DE FILTRAGE (Élimination de ligne)

- **DP**

La valeur (en millimètres à l'échelle de la carte) à utiliser pour élaguer les lignes. Cette opération consiste à éliminer les vertex superflus. La valeur exprimée représente de façon générale l'écart entre les données originales et les données de sortie. L'algorithme de Douglas-Peucker est utilisé lors de cette opération. La valeur **0.0** signifie aucun traitement.

Exemple: Pour une ligne AB (figure 5), les deux extrémités sont connectées par une ligne droite. La perpendiculaire est calculée pour chacun des vertex. Si une des perpendiculaires excède la tolérance spécifiée, le vertex qui a la plus grande perpendiculaire est utilisé comme second point pour subdiviser la ligne originale (figure 6). La perpendiculaire est calculée de nouveau pour chacun des vertex. Si aucune des perpendiculaires excède la tolérance spécifiée, tous les vertex entre les noeuds sont alors éliminés. Cette procédure est utilisée jusqu'à ce que tous les points possibles soient éliminés (figure 7).

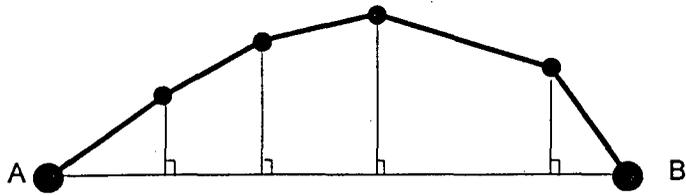


Figure 5 Exemple 1 de Douglas-Peucker

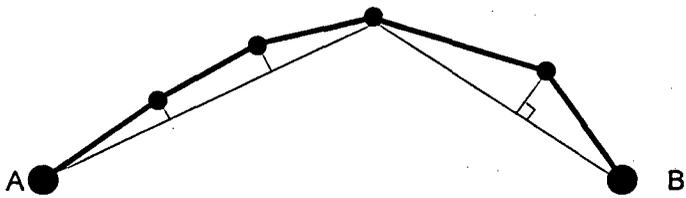


Figure 6 Exemple 2 de Douglas-Peucker

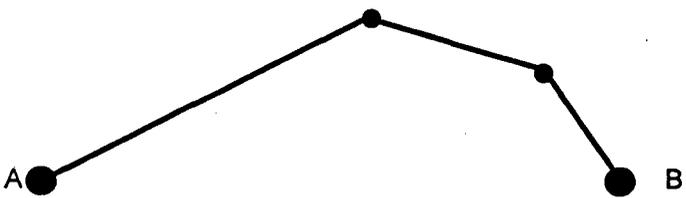
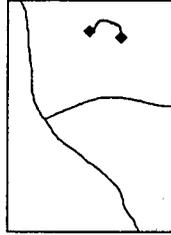


Figure 7 Exemple 3 de Douglas-Peucker

- **2NP**

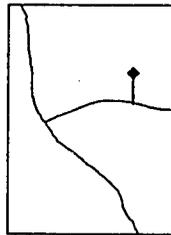
Longueur maximale des lignes à éliminer qui comportent deux (2) noeuds pendants. Toutes les lignes possédant 2 noeuds pendants et dont la longueur est inférieure à la longueur spécifiée seront éliminées. La valeur **0.0** signifie aucun traitement.



**Figure 8** Exemple de 2 noeuds pendants

- **1NP**

Longueur maximale des lignes à éliminer qui comportent un seul (1) noeud pendent. Toutes les lignes possédant 1 noeud pendent et dont la longueur est inférieure à la longueur spécifiée seront éliminées. Le noeud connecté doit être connecté à au moins deux autres lignes. La valeur **0.0** signifie aucun traitement.



**Figure 9** Exemple de 1 noeud pendent

- **2NC**

Longueur maximale des lignes à éliminer qui comportent 2 noeuds connectés. Toutes les lignes possédant 2 noeuds connectés, dont la longueur est inférieure à la longueur spécifiée et qui ne sont pas dans le prolongement d'autres lignes seront éliminées (voir **A\_2NC**). Ces lignes doivent être connectées à au moins 2 autres lignes à chacune des extrémités. La valeur 0.0 signifie aucun traitement.

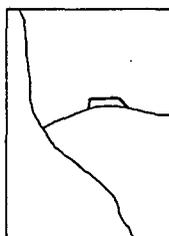


Figure 10 Exemple de 2 noeuds connectés

- **A\_2NC**

Angle de déflexion minimal requis pour éliminer les lignes **2NC**. Cet angle est formé par le prolongement du dernier segment avec les autres lignes connectées aux extrémités. Noter que plus l'angle est grand, moins nombreuses seront les lignes éliminées.

- **ILOT**

Longueur maximale des lignes à éliminer formant un îlot. Tous les îlots dont la longueur est inférieure à la longueur spécifiée seront éliminés. Un îlot est formé par une ligne refermée sur elle-même. La valeur 0.0 signifie aucun traitement.



Figure 11 Exemple d'un îlot

- **ILOT\_C**

Longueur maximale des lignes à éliminer formant un îlot connecté à une ou plusieurs lignes. Tous les îlots connectés dont la longueur est inférieure à la longueur spécifiée seront éliminés. La valeur **0.0** signifie aucun traitement.

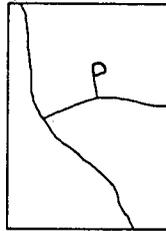


Figure 12 Exemple d'un îlot connecté à une autre ligne

### OPTIONS DE LISSAGE

- **LIS**

Pourcentage de vertex conservés lors du lissage d'une ligne. (Valeur comprise entre 0.001 et 1.0) La valeur **1.0** signifie aucun traitement.

- **N\_V**

Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit être activé. Un noeud virtuel est un vertex qui doit demeurer fixe lors du traitement de lissage. Celui-ci ne sera donc ni éliminé ni repositionné. Un vertex est considéré comme étant un noeud virtuel lorsque celui-ci rencontre les critères définis par les options **L\_V** et **A\_V**.

- **L\_V**

Longueur minimale d'un des deux segments du vertex pour considérer un noeud virtuel. Plus **L\_V** est grand, plus le nombre de noeuds virtuels créés sera faible.

- **A\_V**

Angle intérieur maximal formé par les 2 segments d'un vertex afin de déterminer un noeud virtuel. Plus **A\_V** est grand, plus le nombre de noeuds virtuels créés sera grand.

- **DI\_V**

Distance minimale à considérer lors de la création de noeuds virtuels. Pour un vertex donné satisfaisant aux conditions de **L\_V** et **A\_V**, et qu'un noeud virtuel existe à l'intérieur de ce rayon de recherche sur une même ligne, aucun noeud virtuel ne sera créé. Plus **DI\_V** est grand, plus le nombre de noeuds virtuels créés sera faible.

### 7.3.4 EXEMPLE D'UTILISATION DU PROGRAMME GEN\_MTQ OPTION PHASE 1

Arc: &r gen\_mtq 20k 50k he101c he101c\_50k -ph=1

Debut des traitements de la phase 1 a 12.18.49 le 27/05/1997

Initialisation des parametres en cours ...

Lecture du fichier de configuration en cours...

Les variables sont assignees.

Lecture de la table tables.tab en cours...

Recherche de la table correspondant aux parametres 20k 50k

Validation de la table de parametres 20k\_50k.tab en cours...

Creation de la table des lignes temporaire INFO

Chargement de la table des lignes en cours ...

Fin du chargement de la table

Creation de la table des textes temporaire INFO

Chargement de la table des textes en cours ...

Fin du chargement de la table

Creation de la couverture de sortie "he101c\_50k" en cours...

Creation de la couverture de sortie "he101c\_50k" (lignes) terminee

Changement de la taille des textes en cours...

Creation de la couverture de sortie "he101c\_50k\_t" (textes) terminee

Creation de la couverture de sortie "he101c\_50k" terminee

Creation du fichier de statistiques des lignes de "he101c\_50k" en cours...

Lecture de la table "/data/mtq/table/20k\_50k.tab" en cours...

Ecriture du fichier temporaire pour nouveau code en cours...

Lecture du fichier de statistiques en cours...

Aucun changement de code pour la couverture "he101c\_50k"

Destruction des pseudo-noeuds en cours...

10 pseudo-noeuds detruits

Creation du fichier de statistiques des lignes de "he101c\_50k" en cours...

Creation du fichier de statistiques final des lignes de "he101c\_50k" en cours...

Statistiques des lignes pour la couverture he101c\_50k

#### INDICATIF OCCURRENCES

23020	51
23021	34

(Voir he101c\_50k\_lignes\_stat.log du repertoire des donnees)

Creation du fichier de statistiques des textes de "he101c\_50k\_t" en cours...  
Lecture de la table "/data/mtq/table/20k\_50k.tab" en cours...  
Ecriture du fichier temporaire pour nouveau code en cours...  
Lecture du fichier de statistiques en cours...  
Aucun changement de code pour la couverture "he101c\_50k\_t"  
Creation du fichier de statistiques des textes de "he101c\_50k\_t" en cours...  
Creation du fichier de statistiques final des textes de "he101c\_50k\_t" en cours...

Statistiques des textes pour la couverture he101c\_50k\_t

INDICATIF OCCURRENCES

23220 51

(Voir he101c\_50k\_t\_textes\_stat.log du repertoire des donnees)

Fin des traitements de la phase 1 a 12.20.54 le 27/05/1997

Arc:

### **7.3.5 EXEMPLE D'UTILISATION DU PROGRAMME GEN\_MTQ OPTION PHASE 2**

---

Arc: &r gen\_mtg 20k 50k he201\_50k he201\_50k\_g -ph=2

Debut des traitements de la phase 2 a 12.31.11 le 27/05/1997

Initialisation des parametres en cours ...  
Lecture du fichier de configuration en cours...

Les variables sont assignees.

Lecture de la table tables.tab en cours...  
Recherche de la table correspondant aux parametres 20k 50k  
Validation de la table de parametres 20k\_50k.tab en cours...  
Creation du fichier ID de "he201\_50k" en cours...  
Creation du fichier ASCII temporaire en cours...  
Creation du fichier ASCII generalise en cours...  
Creation de la couverture generalisee "he201\_50k\_g" en cours...  
Assignation de l'indicatif de la couverture generalisee en cours...  
Creation de la table temporaire INFO

Chargement du fichier ID dans la table en cours ...

Fin du chargement de la table

Creation du fichier de statistiques des lignes de "he201\_50k\_g" en cours...

Creation du fichier de statistiques final des lignes de "he201\_50k\_g" en cours...

Statistiques des lignes pour la couverture he201\_50k\_g

INDICATIF	OCCURRENCES
02010001001	99
02010003021	3490
02010003025	83
02010025031	19
02010025041	193
02010025045	5
02020001000	98
02600001000	30
02600002000	50
02600015000	314
02600030001	18

(Voir he201\_50k\_g\_lignes\_stat.log du repertoire des donnees)

Fin des traitements de la phase 2 a 12.34.43 le 27/05/1997

Arc:

## 7.4 PROGRAMME *DETEC\_LIGNES*

---

Ce programme permet de détecter les lignes trop rapprochées selon un critère de rayon de recherche. L'exemple suivant présente les arguments de la commande lorsque celle-ci est exécutée sans aucun argument:

*Arc: &r detec\_lignes*

*Usage: detec\_lignes <couverture> <echelle> <rayon\_mm> <distance\_min\_mm>*

*Exemple: detec\_lignes he202\_500k 500000 0.5 2.0*

*Le rayon de recherche est exprimé en millimètre*

*La distance minimale (en millimètre) requise pour considérer une occurrence*

*NOTE: Si vous désirez ne pas utiliser le paramètre distance\_min\_mm, entrez la valeur 0*

L'argument <couverture> représente la **couverture** à traiter.

L'argument <echelle> représente l'échelle de traçage de la **couverture**.

L'argument <rayon\_mm> représente le rayon de recherche des lignes trop rapprochées.

L'argument <distance\_min\_mm> représente la distance minimale requise pour considérer une occurrence.

L'algorithme de recherche utilise un échantillonnage de points. Ceux-ci sont sélectionnés de la façon suivante:

- densification des vertex équivalent à six fois le rayon de recherche;
- création de point au centre de chacun des segments de ligne;
- recherche d'un ou plusieurs autres points selon le critère de rayon de recherche à partir de chacun des points;
- création d'une table INFO contenant les points détectés ainsi que leurs distances par rapports aux autres points;
- création d'une **couverture** de polygone à partir des points associés à la table INFO;
- détection des segments de ligne par superposition de **couverture** (lignes et polygones);
- élimination de certains points tels que les intersections de ligne (jonction de route);
- création d'une **couverture** de polygones finale;
- identification des lignes trop rapprochées par le biais du champ **DETEC\_LIGNES** dans la **couverture** traitée.

Une occurrence est identifiée lorsque la longueur totale des segments d'une ligne à l'intérieur des polygones est plus grande que l'argument <distance\_min\_mm>.

Lorsqu'il y a détection de lignes trop rapprochées, une **couverture** de polygones est générée et porte le nom de la **couverture** traitée suivi de **\_EP**.

Pour visualiser une couverture, utiliser le programme **ae**.

#### 7.4.1 EXEMPLE D'UTILISATION DU PROGRAMME DETEC\_LIGNES

Arc: &r detec\_lignes c\_500k 500000 0.5 2.0

Debut de la detection des elements trop rapproches a 12.47.39 le 27/05/1997

Initialisation des parametres en cours ...

Lecture du fichier de configuration en cours...

Les variables sont assignees.

Initialisation en cours...

Traitement de la couverture C\_500K en cours...

Creation de la table INFO de distance en cours...

Creation des polygones en cours...

Elimination des enregistrements non-requis en cours...

27 lignes detectees!!!

Creation de la couverture de polygone C\_500K\_EP en cours...

Traitement de la couverture C\_500K termine

Fin de la detection des elements trop rapproches 12.57.25 le 27/05/1997

Arc:

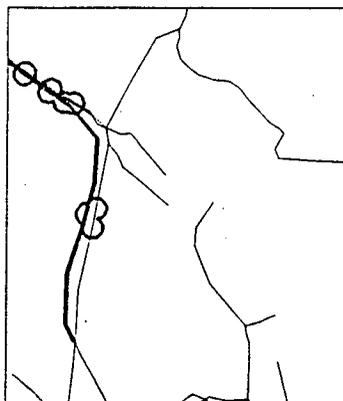
#### 7.5 PROGRAMME AE

Ce programme permet de visualiser les lignes identifiées comme étant trop rapprochées à l'aide du module **ARCEDIT**. L'exemple suivant présente les arguments de la commande lorsque celle-ci est exécutée sans aucun argument:

Usage: **ae** <couverture>

L'argument <couverture> représente la **couverture** à visualiser.

Les lignes détectées sont affichées en rouge alors que la couverture de polygone est affichée en vert (figure 13).



**Figure 13** Exemple de détection de lignes trop rapprochées

Ce programme peut être utilisé pour éditer une couverture sans lignes détectées. Le message suivant est alors affiché:

*Aucune couverture de polygone existante (NOM\_DE\_LA\_COUVERTURE\_EP)*

#### **7.5.1** EXEMPLE D'UTILISATION DU PROGRAMME AE

---

*Arc: &r ae c\_500k*

*Coverage has no COGO attributes*

*27 ligne(s) trop rapprochee(s) detectee(s)*

*Arccedit:*

```

# Table 20k 50k
# =====
# DESCRIPTIONS DES PARAMETRES
#
# Le caractere "#" permet de mettre des lignes en commentaire. Ces lignes sont alors ignorees.
#
# Le caractere d'espace est utilise comme separateur de champ.
#
# La premiere ligne qui ne debute pas par le caractere "#" doit contenir la chaine
# de caractere Echelles= suivie d'un caractere d'espace et des valeurs associees
# aux echelles d'entree et sortie. Ces valeurs doivent etre exprimees en nombres entiers.
# Exemple: Echelles= 20000 50000
#
# La ligne suivante doit contenir la chaine de caractere Resolution= suivie d'un
# caractere d'espace et de l'unite de resolution.
# Les valeurs possibles sont les suivantes: M (metre)
#                                     DM (decimetre)
#                                     CM (centimetre)
#                                     MM (millimetre)
# Exemple: Resolution= M
#
# NOTE1: Les valeurs associees a une longueur sont toujours exprimees en milimetres
#        a l'echelle de la carte (valeur REEL, ex.: 1.0).
# NOTE2: Les valeurs associees a un angle sont toujours exprimees en degres decimaux
#        (valeur REEL, ex.: 25.0).
#
# Les options suivantes sont utilisees par la AML principale "gen_mtg" lors de la PHASE1.
# =====
#
# TYPE = Type d'entite a transferer dans la couverture de sortie
#        Les types valides sont les suivants: L pour entite lineaire, T pour textuelle,
#        P pour ponctuelle ou les combinaisons suivantes: * pour toutes, LP, LT et PT.
#
# INDICATIF ENTREE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
# INDICATIF SORTIE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
#                   Si l'indicatif de sortie est egal a l'indicatif d'entree, il est
#                   possible d'utiliser le caractere "@" pour l'indiquer.
#
# PN = Indiquer par O (oui) ou N (non) si les pseudo-noeuds doivent etre enleves.
#      Un pseudo-noeud represente la jonction de deux lignes connectees ayant
#      la meme classe. En eliminant le pseudo-noeud, les deux lignes seront fusionnees
#      en une seule ligne.
#      NOTE1: Cette option est utilisee par le programme C++ "genligne" lors du filtrage.
#             (PHASE2 de la AML principale "gen_mtg")
#      NOTE2: Il est a noter que la majorite des pseudo-noeuds sont elimines de facon systematique
#             lors de la phase1 (Selection des elements selon les echelles d'entree et sortie).
#             Cette option permet d'eliminer en plus les pseudo-noeuds creer par ARC/INFO
#             pour les lignes composees de plus de 500 vertex. (contrainte ARC/INFO)
#
# RECL = Longueur maximale des elements a reclassifier. Toutes les lignes plus courtes que la
#        longueur specifiee seront reclassifiees. le caractere "-" signifie que l'element doit
#        etre reclassifie sans aucun critere de longueur. La valeur "0.0" permet de ne reclassifier
#        aucun element. Les elements reclassifies prennent l'indicatif d'un des elements adjacents.
#        Exemple: Les ponts sont reclassifies selon l'indicatif d'une des routes
#        connectee au pont.
#        NOTE: Cette option est utilisee par la AML "gen_mtg" lors de la phase1.
#
# Les options suivantes sont utilisees par le programme C++ "genligne".
# =====
# OPTIONS DE FILTRAGE, PHASE2 de la AML "gen_mtg".
#
# DP = La valeur (en milimetres a l'echelle de la carte) a utiliser pour elaguer les lignes.
#      Cette operation consiste a eliminer les vertex superflus. La valeur exprimee represente
#      de facon generale l'eccart entre les donnees originales et les donnees de sortie.
#      L'algorithme de Douglas-Peucker est utilise lors de cette operation.
#      La valeur "0.0" signifie aucun traitement.

```

```

#
# 2NP = Longueur maximale des lignes a eliminer qui comportent 2 noeuds pendants.
# Toutes les lignes possedant 2 noeuds pendants et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# La valeur "0.0" signifie aucun traitement.
#
# 1NP = Longueur maximale des lignes a eliminer qui comportent 1 seul noeud pendant.
# Toutes les lignes possedant 1 noeud pendant et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# Le noeud connecte doit etre connecte a au moins deux autres lignes.
# La valeur "0.0" signifie aucun traitement.
#
# 2NC = Longueur maximale des lignes a eliminer qui comportent 2 noeuds connectes.
# Toutes les lignes possedant 2 noeuds connectes, dont la longueur est inferieure a la
# longueur specifiee et qui ne sont pas dans le prolongement d'autres lignes
# seront eliminees (voir A_2NC).
# Ces lignes doivent etre connectees a au moins 2 autres lignes a chacune des extremités.
# La valeur "0.0" signifie aucun traitement.
#
# A_2NC = Angle de deflexion minimal requis pour eliminer les lignes 2NC.
# Cet angle est forme par le prolongement du dernier segment avec les autres
# lignes connectees aux extremités. Noter que plus l'angle est grand, moins nombreuses
# seront les lignes eliminees.
#
# ILOT = Longueur maximale des lignes a eliminer formant un ilot.
# Tous les ilots dont la longueur est inferieure a la longueur specifiee seront elimines.
# Un ilot est forme par une ligne refermee sur elle-meme.
# La valeur "0.0" signifie aucun traitement.
#
# ILOT_C = Longueur maximale des lignes a eliminer formant un ilot connecte a une ou plusieurs lignes.
# Tous les ilots connectes dont la longueur est inferieure a la longueur specifiee seront
# elimines.
# La valeur "0.0" signifie aucun traitement.
#
# =====
# OPTIONS DE LISSAGE, PHASE2 de la AML "gen_mtg".
#
# LIS = Pourcentage de vertex conserves lors du lissage d'une ligne.
# (Valeur comprise entre 0.001 et 1.0)
# La valeur "1.0" signifie aucun traitement.
#
# N_V = Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit etre active.
#
# L_V = Longueur minimale d'un des deux segments d'un vertex pour considerer un noeud virtuel.
# Plus L_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# A_V = Angle interieur maximal forme par les 2 segments d'un vertex afin de determiner
# un noeud virtuel.
# Plus A_V est grand, plus le nombre de noeuds virtuels crees sera grand.
#
# DI_V = Distance minimale a considerer lors de la creation de noeuds virtuels.
# Pour un vertex donne satisfaisant aux conditions de L_V et A_V, et qu'un noeud virtuel
# existe a l'interieur de ce rayon de recherche sur une meme ligne, aucun noeud virtuel
# ne sera cree.
# Plus DI_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# =====
# Echelle_entree Echelle_sortie
# Echelles= 20000 50000
# Unite de resolution
# Resolution= M
# =====
#TYPE INDICATIF INDICATIF PN RECL DP 2NP 1NP 2NC A_2NC ILOT ILOT_C LIS N_V L_V A_V DI_V
# ENTREE SORTIE
# =====
L 02010000006 02010000006 O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0
L 02010001001 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0
L 02010001002 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0
L 02010025011 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 10.0 170.0 10.0
L 02010025015 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 10.0 170.0 10.0
L 02010025021 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 10.0 170.0 10.0
L 02010025025 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 10.0 170.0 10.0
L 02010025031 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 8.0 170.0 8.0
L 02010025035 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 8.0 170.0 8.0
L 02010025041 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0
L 02010025045 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0
L 02010025051 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.8 O 5.0 120.0 5.0

```

L	02010025055 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02010003011 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.8	O	2.0	140.0	2.0
L	02010003015 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.8	O	2.0	140.0	2.0
L	02010003021 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.8	O	0.1	160.0	0.1
L	02010003025 @	O	0.0	0.1	10.0	8.0	10.0	25.0	20.0	10.0	0.8	O	0.1	160.0	0.1
L	02010010000 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02010010005 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02020001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02030001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600001001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600002000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600004000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600010000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600015000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	1.0	O	5.0	120.0	5.0
L	02600030001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
L	02600030002 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.8	O	5.0	120.0	5.0
# Donnees cadastrales															
L	23020 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	10.0	1.0	O	5.0	120.0	5.0
L	23021 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	10.0	1.0	O	5.0	120.0	5.0
T	23220 @	O	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	O	5.0	120.0	5.0

```

# Table 20k 125k
#-----
# DESCRIPTIONS DES PARAMETRES
#
# Le caractere "#" permet de mettre des lignes en commentaire. Ces lignes sont alors ignorees.
#
# Le caractere d'espace est utilise comme separateur de champ.
#
# La premiere ligne qui ne debute pas par le caractere "#" doit contenir la chaine
# de caractere Echelles= suivie d'un caractere d'espace et des valeurs associees
# aux echelles d'entree et sortie. Ces valeurs doivent etre exprimees en nombres entiers.
# Exemple: Echelles= 20000 50000
#
# La ligne suivante doit contenir la chaine de caractere Resolution= suivie d'un
# caractere d'espace et de l' unite de resolution.
# Les valeurs possibles sont les suivantes: M (metre)
#                                     DM (decimetre)
#                                     CM (centimetre)
#                                     MM (millimetre)
# Exemple: Resolution= M
#
# NOTE1: Les valeurs associees a une longueur sont toujours exprimees en milimetres
#        a l'echelle de la carte (valeur REEL, ex.: 1.0).
# NOTE2: Les valeurs associees a un angle sont toujours exprimees en degres decimaux
#        (valeur REEL, ex.: 25.0).
#
# Les options suivantes sont utilisees par la AML principale "gen_mtg" lors de la PHASE1.
#-----
# TYPE = Type d'entite a transferer dans la couverture de sortie
#        Les types valides sont les suivants: L pour entite lineaire, T pour textuelle,
#        P pour ponctuelle ou les combinaisons suivantes: * pour toutes, LP, LT et PT.
#
# INDICATIF ENTREE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
# INDICATIF SORTIE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
#                   Si l'indicatif de sortie est egal a l'indicatif d'entree, il est
#                   possible d'utiliser le caractere "@" pour l'indiquer.
#
# PN = Indiquer par O (oui) ou N (non) si les pseudo-noeuds doivent etre enleves.
#      Un pseudo-noeud represente la jonction de deux lignes connectees ayant
#      la meme classe. En eliminant le pseudo-noeud, les deux lignes seront fusionnees
#      en une seule ligne.
#      NOTE1: Cette option est utilisee par le programme C++ "genligne" lors du filtrage.
#            (PHASE2 de la AML principale "gen_mtg")
#      NOTE2: Il est a noter que la majorite des pseudo-noeuds sont elimines de facon systematique
#            lors de la phase1 (Selection des elements selon les echelles d'entree et sortie).
#            Cette option permet d'eliminer en plus les pseudo-noeuds creer par ARC/INFO
#            pour les lignes composees de plus de 500 vertex. (contrainte ARC/INFO)
#
# RECL = Longueur maximale des elements a reclassifier. Toutes les lignes plus courtes que la
#        longueur specifiee seront reclassifiees. le caractere "-" signifie que l'element doit
#        etre reclassifie sans aucun critere de longueur. La valeur "0.0" permet de ne reclassifier
#        aucun element. Les elements reclassifies prennent l'indicatif d'un des elements adjacents.
#        Exemple: Les ponts sont reclassifies selon l'indicatif d'une des routes
#        connectee au pont.
#        NOTE: Cette option est utilisee par la AML "gen_mtg" lors de la phase1.
#
# Les options suivantes sont utilisees par le programme C++ "genligne".
#-----
# OPTIONS DE FILTRAGE, PHASE2 de la AML "gen_mtg".
#
# DP = La valeur (en milimetres a l'echelle de la carte) a utiliser pour elaguer les lignes.
#      Cette operation consiste a eliminer les vertex superflus. La valeur exprimee represente
#      de facon generale l'eccart entre les donnees originales et les donnees de sortie.
#      L'algorithme de Douglas-Peucker est utilise lors de cette operation.
#      La valeur "0.0" signifie aucun traitement.
#
# 2NP = Longueur maximale des lignes a eliminer qui comportent 2 noeuds pendants.

```

```

#      Toutes les lignes possedant 2 noeuds pendants et dont la longueur est inferieure a la
#      longueur specifiee seront eliminees.
#      La valeur "0.0" signifie aucun traitement.
#
# INP = Longueur maximale des lignes a eliminer qui comportent 1 seul noeud pendent.
#      Toutes les lignes possedant 1 noeud pendent et dont la longueur est inferieure a la
#      longueur specifiee seront eliminees.
#      Le noeud connecte doit etre connecte a au moins deux autres lignes.
#      La valeur "0.0" signifie aucun traitement.
#
# 2NC = Longueur maximale des lignes a eliminer qui comportent 2 noeuds connectes.
#      Toutes les lignes possedant 2 noeuds connectes, dont la longueur est inferieure a la
#      longueur specifiee et qui ne sont pas dans le prolongement d'autres lignes
#      seront eliminees (voir A_2NC).
#      Ces lignes doivent etre connectees a au moins 2 autres lignes a chacune des extremités.
#      La valeur "0.0" signifie aucun traitement.
#
# A_2NC = Angle de deflexion minimal requis pour eliminer les lignes 2NC.
#      Cet angle est forme par le prolongement du dernier segment avec les autres
#      lignes connectees aux extremités. Noter que plus l'angle est grand, moins nombreuses
#      seront les lignes eliminees.
#
# ILOT = Longueur maximale des lignes a eliminer formant un ilot.
#      Tous les ilots dont la longueur est inferieure a la longueur specifiee seront elimines.
#      Un ilot est forme par une ligne refermee sur elle-meme.
#      La valeur "0.0" signifie aucun traitement.
#
# ILOT_C = Longueur maximale des lignes a eliminer formant un ilot connecte a une ou plusieurs lignes.
#      Tous les ilots connectes dont la longueur est inferieure a la longueur specifiee seront
#      elimines.
#      La valeur "0.0" signifie aucun traitement.
#
# =====
# OPTIONS DE LISSAGE, PHASE2 de la AML "gen_mtq".
#
# LIS   = Pourcentage de vertex conserves lors du lissage d'une ligne.
#      (Valeur comprise entre 0.001 et 1.0)
#      La valeur "1.0" signifie aucun traitement.
#
# N_V   = Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit etre active.
#
# L_V   = Longueur minimale d'un des deux segments d'un vertex pour considerer un noeud virtuel.
#      Plus L_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# A_V   = Angle interieur maximal forme par les 2 segments d'un vertex afin de determiner
#      un noeud virtuel.
#      Plus A_V est grand, plus le nombre de noeuds virtuels crees sera grand.
#
# DI_V  = Distance minimale a considerer lors de la creation de noeuds virtuels.
#      Pour un vertex donne satisfaisant aux conditions de L_V et A_V, et qu'un noeud virtuel
#      existe a l'interieur de ce rayon de recherche sur une meme ligne, aucun noeud virtuel
#      ne sera cree.
#      Plus DI_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# =====
# Echelle_entree Echelle_sortie
# Echelles= 20000 125000
# Unite de resolution
# Resolution= M
# =====
#TYPE INDICATIF   INDICATIF PN RECL DP  2NP INP  2NC  A_2NC ILOT ILOT_C LIS  N_V  L_V  A_V  DI_V
#      ENTREE      SORTIE
# =====
L 02010000006 02010000006 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010001001 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010001002 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025011 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025015 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025021 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025025 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025031 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 8.0 170.0 8.0
L 02010025035 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 8.0 170.0 8.0
L 02010025041 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025045 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025051 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025055 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010003011 @ 0 0.0 0.1 7.0 3.0 10.0 25.0 20.0 10.0 0.7 0 2.0 140.0 2.0

```

L	02010003015 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.7	O	2.0	140.0	2.0
L	02010003021 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.7	O	0.1	160.0	0.1
L	02010003025 @	O	0.0	0.1	10.0	8.0	10.0	25.0	20.0	10.0	0.7	O	0.1	160.0	0.1
L	02010010000 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
L	02010010005 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
L	02020001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
L	02030001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600001001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600002000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600004000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600010000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600015000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	1.0	O	5.0	120.0	5.0
L	02600030001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600030002 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
#	Donnees cadastrales														
L	23020 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	0.0	1.0	O	5.0	120.0	5.0
L	23021 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	0.0	1.0	O	5.0	120.0	5.0
T	23220 @	N	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	O	5.0	120.0	5.0

```

# Table 20k 500k
#-----
# DESCRIPTIONS DES PARAMETRES
#
# Le caractere "#" permet de mettre des lignes en commentaire. Ces lignes sont alors ignorees.
#
# Le caractere d'espacement est utilise comme separateur de champ.
#
# La premiere ligne qui ne debute pas par le caractere "#" doit contenir la chaine
# de caractere Echelles= suivie d'un caractere d'espacement et des valeurs associees
# aux echelles d'entree et sortie. Ces valeurs doivent etre exprimees en nombres entiers.
# Exemple: Echelles= 20000 50000
#
# La ligne suivante doit contenir la chaine de caractere Resolution= suivie d'un
# caractere d'espacement et de l' unite de resolution.
# Les valeurs possibles sont les suivantes: M (metre)
#                                     DM (decimetre)
#                                     CM (centimetre)
#                                     MM (millimetre)
# Exemple: Resolution= M
#
# NOTE1: Les valeurs associees a une longueur sont toujours exprimees en milimetres
#        a l'echelle de la carte (valeur REEL, ex.: 1.0).
# NOTE2: Les valeurs associees a un angle sont toujours exprimees en degres decimaux
#        (valeur REEL, ex.: 25.0).
#
# Les options suivantes sont utilisees par la AML principale "gen_mtg" lors de la PHASE1.
#-----
#
# TYPE = Type d'entite a transferer dans la couverture de sortie
#        Les types valides sont les suivants: L pour entite lineaire, T pour textuelle,
#        P pour ponctuelle ou les combinaisons suivantes: * pour toutes, LP, LT et PT.
#
# INDICATIF ENTREE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
# INDICATIF SORTIE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
#                   Si l'indicatif de sortie est egal a l'indicatif d'entree, il est
#                   possible d'utiliser le caractere "@" pour l'indiquer.
#
# PN = Indiquer par O (oui) ou N (non) si les pseudo-noeuds doivent etre enleves.
#      Un pseudo-noeud represente la jonction de deux lignes connectees ayant
#      la meme classe. En eliminant le pseudo-noeud, les deux lignes seront fusionnees
#      en une seule ligne.
#      NOTE1: Cette option est utilisee par le programme C++ "genligne" lors du filtrage.
#             (PHASE2 de la AML principale "gen_mtg")
#      NOTE2: Il est a noter que la majorite des pseudo-noeuds sont elimines de facon systematique
#             lors de la phasel (Selection des elements selon les echelles d'entree et sortie).
#             Cette option permet d'eliminer en plus les pseudo-noeuds creer par ARC/INFO
#             pour les lignes composees de plus de 500 vertex. (contrainte ARC/INFO)
#
# RECL = Longueur maximale des elements a reclassifier. Toutes les lignes plus courtes que la
#        longueur specifiee seront reclassifiees. le caractere "-" signifie que l'element doit
#        etre reclassifie sans aucun critere de longueur. La valeur "0.0" permet de ne reclassifier
#        aucun element. Les elements reclassifies prennent l'indicatif d'un des elements adjacents.
#        Exemple: Les ponts sont reclassifies selon l'indicatif d'une des routes
#                 connectee au pont.
#        NOTE: Cette option est utilisee par la AML "gen_mtg" lors de la phasel.
#
# Les options suivantes sont utilisees par le programme C++ "genligne".
#-----
# OPTIONS DE FILTRAGE, PHASE2 de la AML "gen_mtg".
#
# DP = La valeur (en milimetres a l'echelle de la carte) a utiliser pour elaguer les lignes.
#      Cette operation consiste a eliminer les vertex superflus. La valeur exprimee represente
#      de facon generale l'eccart entre les donnees originales et les donnees de sortie.
#      L'algorithme de Douglas-Peucker est utilise lors de cette operation.
#      La valeur "0.0" signifie aucun traitement.
#
# 2NP = Longueur maximale des lignes a eliminer qui comportent 2 noeuds pendants.

```

```

#      Toutes les lignes possedant 2 noeuds pendants et dont la longueur est inferieure a la
#      longueur specifiee seront eliminees.
#      La valeur "0.0" signifie aucun traitement.
#
# 1NP = Longueur maximale des lignes a eliminer qui comportent 1 seul noeud pendant.
#      Toutes les lignes possedant 1 noeud pendant et dont la longueur est inferieure a la
#      longueur specifiee seront eliminees.
#      Le noeud connecte doit etre connecte a au moins deux autres lignes.
#      La valeur "0.0" signifie aucun traitement.
#
# 2NC = Longueur maximale des lignes a eliminer qui comportent 2 noeuds connectes.
#      Toutes les lignes possedant 2 noeuds connectes, dont la longueur est inferieure a la
#      longueur specifiee et qui ne sont pas dans le prolongement d'autres lignes
#      seront eliminees (voir A_2NC).
#      Ces lignes doivent etre connectees a au moins 2 autres lignes a chacune des extremités.
#      La valeur "0.0" signifie aucun traitement.
#
# A_2NC = Angle de deflexion minimal requis pour eliminer les lignes 2NC.
#      Cet angle est forme par le prolongement du dernier segment avec les autres
#      lignes connectees aux extremités. Noter que plus l'angle est grand, moins nombreuses
#      seront les lignes eliminees.
#
# ILOT = Longueur maximale des lignes a eliminer formant un ilot.
#      Tous les ilots dont la longueur est inferieure a la longueur specifiee seront elimines.
#      Un ilot est forme par une ligne refermee sur elle-meme.
#      La valeur "0.0" signifie aucun traitement.
#
# ILOT_C = Longueur maximale des lignes a eliminer formant un ilot connecte a uné ou plusieurs lignes.
#      Tous les ilots connectes dont la longueur est inferieure a la longueur specifiee seront
#      elimines.
#      La valeur "0.0" signifie aucun traitement.
#
=====
# OPTIONS DE LISSAGE, PHASE2 de la AML "gen_mtg".
#
# LIS      = Pourcentage de vertex conserves lors du lissage d'une ligne.
#           (Valeur comprise entre 0.001 et 1.0)
#           La valeur "1.0" signifie aucun traitement.
#
# N_V      = Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit etre active.
#
# L_V      = Longueur minimale d'un des deux segments d'un vertex pour considerer un noeud virtuel.
#           Plus L_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# A_V      = Angle interieur maximal forme par les 2 segments d'un vertex afin de determiner
#           un noeud virtuel.
#           Plus A_V est grand, plus le nombre de noeuds virtuels crees sera grand.
#
# DI_V     = Distance minimale a considerer lors de la creation de noeuds virtuels.
#           Pour un vertex donne satisfaisant aux conditions de L_V et A_V, et qu'un noeud virtuel
#           existe a l'interieur de ce rayon de recherche sur une meme ligne, aucun noeud virtuel
#           ne sera cree.
#           Plus DI_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
=====
# Echelle_entree Echelle_sortie
Echelles= 20000 500000
# Unite de resolution
Resolution= M
#
#TYPE INDICATIF   INDICATIF PN RECL DP  2NP 1NP  2NC  A_2NC ILOT ILOT_C LIS  N_V  L_V  A_V  DI_V
#  ENTREE        SORTIE
#
L 02010000006 02010000006 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010001001 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010001002 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025011 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025015 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025021 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025025 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 10.0 170.0 10.0
L 02010025031 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 3.0 170.0 3.0
L 02010025035 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 3.0 170.0 3.0
L 02010025041 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 3.0 120.0 3.0
L 02010025045 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 3.0 120.0 3.0
L 02010025051 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
L 02010025055 @ 0 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 0 5.0 120.0 5.0
# L 02010003011 @ 0 0.0 0.1 7.0 3.0 10.0 25.0 20.0 10.0 0.7 0 2.0 140.0 2.0

```

# L	02010003015 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.7	O	2.0	140.0	2.0
L	02010003021 @	O	0.0	0.1	7.0	3.0	10.0	25.0	20.0	10.0	0.7	O	0.1	160.0	0.1
L	02010003025 @	O	0.0	0.1	10.0	8.0	10.0	25.0	20.0	10.0	0.7	O	0.1	160.0	0.1
# L	02010010000 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
# L	02010010005 @	O	0.0	0.1	10.0	8.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
L	02020001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	140.0	5.0
L	02030001000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600001000 @	O	2.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600001001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600002000 @	O	2.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
# L	02600004000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
L	02600010000 @	O	2.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
# L	02600015000 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
# L	02600030001 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
# L	02600030002 @	O	0.0	0.1	7.0	3.0	5.0	25.0	20.0	10.0	0.7	O	5.0	120.0	5.0
# Donnees cadastrales															
# L	23020 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	0.0	1.0	O	5.0	120.0	5.0
# L	23021 @	O	0.0	0.0	0.0	0.0	0.0	25.0	0.0	0.0	1.0	O	5.0	120.0	5.0
# T	23220 @	N	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	O	5.0	120.0	5.0

```

# Table 50k 125k
# =====
# DESCRIPTIONS DES PARAMETRES
#
# Le caractere "#" permet de mettre des lignes en commentaire. Ces lignes sont alors ignorees.
#
# Le caractere d'espacement est utilise comme separateur de champ.
#
# La premiere ligne qui ne debute pas par le caractere "#" doit contenir la chaine
# de caractere Echelles= suivie d'un caractere d'espacement et des valeurs associees
# aux echelles d'entree et sortie. Ces valeurs doivent etre exprimees en nombres entiers.
# Exemple: Echelles= 20000 50000
#
# La ligne suivante doit contenir la chaine de caractere Resolution= suivie d'un
# caractere d'espacement et de l' unite de resolution.
# Les valeurs possibles sont les suivantes: M (metre)
#                                     DM (decimetre)
#                                     CM (centimetre)
#                                     MM (millimetre)
#
# Exemple: Resolution= M
#
# NOTE1: Les valeurs associees a une longueur sont toujours exprimees en milimetres
#        a l'echelle de la carte (valeur REEL, ex.: 1.0).
# NOTE2: Les valeurs associees a un angle sont toujours exprimees en degres decimaux
#        (valeur REEL, ex.: 25.0).
#
#
# Les options suivantes sont utilisees par la AML principale "gen_mtg" lors de la PHASE1.
# =====
#
# TYPE = Type d'entite a transferer dans la couverture de sortie
#        Les types valides sont les suivants: L pour entite lineaire, T pour textuelle,
#        P pour ponctuelle ou les combinaisons suivantes: * pour toutes, LP, LT et PT.
#
# INDICATIF ENTREE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
# INDICATIF SORTIE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 carac.
# Si l'indicatif de sortie est egal a l'indicatif d'entree, il est
# possible d'utiliser le caractere "@" pour l'indiquer.
#
# PN = Indiquer par O (oui) ou N (non) si les pseudo-noeuds doivent etre enleves.
# Un pseudo-noeud represente la jonction de deux lignes connectees ayant
# la meme classe. En eliminant le pseudo-noeud, les deux lignes seront fusionnees
# en une seule ligne.
# NOTE1: Cette option est utilisee par le programme C++ "genligne" lors du filtrage.
#        (PHASE2 de la AML principale "gen_mtg")
# NOTE2: Il est a noter que la majorite des pseudo-noeuds sont elimines de facon systematique
#        lors de la phase1 (Selection des elements selon les echelles d'entree et sortie).
#        Cette option permet d'eliminer en plus les pseudo-noeuds creer par ARC/INFO
#        pour les lignes composees de plus de 500 vertex. (contrainte ARC/INFO)
#
# RECL = Longueur maximale des elements a reclassifier. Toutes les lignes plus courtes que la
#        longueur specifiee seront reclassifiees. le caractere "-" signifie que l'element doit
#        etre reclassifie sans aucun critere de longueur. La valeur "0.0" permet de ne reclassifier
#        aucun element. Les elements reclassifies prennent l'indicatif d'un des elements adjacents.
#        Exemple: Les ponts sont reclassifies selon l'indicatif d'une des routes
#        connectee au pont.
#        NOTE: Cette option est utilisee par la AML "gen_mtg" lors de la phase1.
#
#
# Les options suivantes sont utilisees par le programme C++ "genligne".
# =====
#
# OPTIONS DE FILTRAGE, PHASE2 de la AML "gen_mtg".
#
# DP = La valeur (en milimetres a l'echelle de la carte) a utiliser pour elaguer les lignes.
# Cette operation consiste a eliminer les vertex superflus. La valeur exprimee represente
# de facon generale l'eccart entre les donnees originales et les donnees de sortie.
# L'algorithme de Douglas-Peucker est utilise lors de cette operation.
# La valeur "0.0" signifie aucun traitement.
#
# 2NP = Longueur maximale des lignes a eliminer qui comportent 2 noeuds pendants.

```

```

# Toutes les lignes possedant 2 noeuds pendants et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# La valeur "0.0" signifie aucun traitement.
#
# 1NP = Longueur maximale des lignes a eliminer qui comportent 1 seul noeud pendant.
# Toutes les lignes possedant 1 noeud pendant et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# Le noeud connecte doit etre connecte a au moins deux autres lignes.
# La valeur "0.0" signifie aucun traitement.
#
# 2NC = Longueur maximale des lignes a eliminer qui comportent 2 noeuds connectes.
# Toutes les lignes possedant 2 noeuds connectes, dont la longueur est inferieure a la
# longueur specifiee et qui ne sont pas dans le prolongement d'autres lignes
# seront eliminees (voir A_2NC).
# Ces lignes doivent etre connectees a au moins 2 autres lignes a chacune des extremités.
# La valeur "0.0" signifie aucun traitement.
#
# A_2NC = Angle de deflexion minimal requis pour eliminer les lignes 2NC.
# Cet angle est forme par le prolongement du dernier segment avec les autres
# lignes connectees aux extremités. Noter que plus l'angle est grand, moins nombreuses
# seront les lignes eliminees.
#
# ILOT = Longueur maximale des lignes a eliminer formant un ilot.
# Tous les ilots dont la longueur est inferieure a la longueur specifiee seront elimines.
# Un ilot est forme par une ligne refermee sur elle-meme.
# La valeur "0.0" signifie aucun traitement.
#
# ILOT_C = Longueur maximale des lignes a eliminer formant un ilot connecte a une ou plusieurs lignes.
# Tous les ilots connectes dont la longueur est inferieure a la longueur specifiee seront
# elimines.
# La valeur "0.0" signifie aucun traitement.
#
=====
# OPTIONS DE LISSAGE, PHASE2 de la AML "gen_mtg".
#
# LIS = Pourcentage de vertex conserves lors du lissage d'une ligne.
# (Valeur comprise entre 0.001 et 1.0)
# La valeur "1.0" signifie aucun traitement.
#
# N_V = Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit etre active.
#
# L_V = Longueur minimale d'un des deux segments d'un vertex pour considerer un noeud virtuel.
# Plus L_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# A_V = Angle interieur maximal forme par les 2 segments d'un vertex afin de determiner
# un noeud virtuel.
# Plus A_V est grand, plus le nombre de noeuds virtuels crees sera grand.
#
# DI_V = Distance minimale a considerer lors de la creation de noeuds virtuels.
# Pour un vertex donne satisfaisant aux conditions de L_V et A_V, et qu'un noeud virtuel
# existe a l'interieur de ce rayon de recherche sur une meme ligne, aucun noeud virtuel
# ne sera cree.
# Plus DI_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
=====
# Echelle_entree Echelle_sortie
Echelles= 50000 125000
# Unite de resolution
Resolution= M
#
#TYPE INDICATIF INDICATIF PN RECL DP 2NP 1NP 2NC A_2NC ILOT ILOT_C LIS N_V L_V A_V DI_V
# ENTREE SORTIE
#
L 935 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 955 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 956 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 998 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 999 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1004 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1005 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1006 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1011 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1012 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1015 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1016 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1020 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0
L 1022 @ O 0.0 0.1 7.0 3.0 5.0 25.0 20.0 10.0 0.7 O 5.0 120.0 5.0

```

```

# Table 50k 500k
#-----
# DESCRIPTIONS DES PARAMETRES
#
# Le caractere "#" permet de mettre des lignes en commentaire. Ces lignes sont alors ignorees.
#
# Le caractere d'espacement est utilise comme separateur de champ.
#
# La premiere ligne qui ne debute pas par le caractere "#" doit contenir la chaine
# de caractere Echelles= suivie d'un caractere d'espacement et des valeurs associees
# aux echelles d'entree et sortie. Ces valeurs doivent etre exprimees en nombres entiers.
# Exemple: Echelles= 20000 50000
#
# La ligne suivante doit contenir la chaine de caractere Resolution= suivie d'un
# caractere d'espacement et de l' unite de resolution.
# Les valeurs possibles sont les suivantes: M (metre)
#                                     DM (decimetre)
#                                     CM (centimetre)
#                                     MM (millimetre)
# Exemple: Resolution= M
#
# NOTE1: Les valeurs associees a une longueur sont toujours exprimees en milimetres
#        a l'echelle de la carte (valeur REEL, ex.: 1.0).
# NOTE2: Les valeurs associees a un angle sont toujours exprimees en degres decimaux
#        (valeur REEL, ex.: 25.0).
#
# Les options suivantes sont utilisees par la AML principale "gen_mtg" lors de la PHASE1.
#-----
# TYPE = Type d'entite a transferer dans la couverture de sortie
#        Les types valides sont les suivants: L pour entite lineaire, T pour textuelle,
#        P pour ponctuelle ou les combinaisons suivantes: * pour toutes, LP, LT et PT.
#
# INDICATIF ENTREE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 caract.
# INDICATIF SORTIE = Indicatif (Graphic Group), champ alphanumerique, maximum de 11 caract.
#                   Si l'indicatif de sortie est egal a l'indicatif d'entree, il est
#                   possible d'utiliser le caractere "@" pour l'indiquer.
#
# PN = Indiquer par O (oui) ou N (non) si les pseudo-noeuds doivent etre enleves.
#      Un pseudo-noeud represente la jonction de deux lignes connectees ayant
#      la meme classe. En eliminant le pseudo-noeud, les deux lignes seront fusionnees
#      en une seule ligne.
#      NOTE1: Cette option est utilisee par le programme C++ "genligne" lors du filtrage.
#            (PHASE2 de la AML principale "gen_mtg")
#      NOTE2: Il est a noter que la majorite des pseudo-noeuds sont elimines de facon systematique
#            lors de la phase1 (Selection des elements selon les echelles d'entree et sortie).
#            Cette option permet d'eliminer en plus les pseudo-noeuds creer par ARC/INFO
#            pour les lignes composees de plus de 500 vertex. (contrainte ARC/INFO)
#
# RECL = Longueur maximale des elements a reclassifier. Toutes les lignes plus courtes que la
#        longueur specifiee seront reclassifiees. le caractere "-" signifie que l'element doit
#        etre reclassifie sans aucun critere de longueur. La valeur "0.0" permet de ne reclassifier
#        aucun element. Les elements reclassifies prennent l'indicatif d'un des elements adjacents.
#        Exemple: Les ponts sont reclassifies selon l'indicatif d'une des routes
#        connectee au pont.
#        NOTE: Cette option est utilisee par la AML "gen_mtg" lors de la phase1.
#
#
# Les options suivantes sont utilisees par le programme C++ "genligne".
#-----
# OPTIONS DE FILTRAGE, PHASE2 de la AML "gen_mtg".
#
# DP = La valeur (en milimetres a l'echelle de la carte) a utiliser pour elaguer les lignes.
#      Cette operation consiste a eliminer les vertex superflus. La valeur exprimee represente
#      de facon generale l'eccart entre les donnees originales et les donnees de sortie.
#      L'algorithme de Douglas-Peucker est utilise lors de cette operation.
#      La valeur "0.0" signifie aucun traitement.
#
# 2NP = Longueur maximale des lignes a eliminer qui comportent 2 noeuds pendants.

```

```

# Toutes les lignes possedant 2 noeuds pendants et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# La valeur "0.0" signifie aucun traitement.
#
# 1NP = Longueur maximale des lignes a eliminer qui comportent 1 seul noeud pendant.
# Toutes les lignes possedant 1 noeud pendant et dont la longueur est inferieure a la
# longueur specifiee seront eliminees.
# Le noeud connecte doit etre connecte a au moins deux autres lignes.
# La valeur "0.0" signifie aucun traitement.
#
# 2NC = Longueur maximale des lignes a eliminer qui comportent 2 noeuds connectes.
# Toutes les lignes possedant 2 noeuds connectes, dont la longueur est inferieure a la
# longueur specifiee et qui ne sont pas dans le prolongement d'autres lignes
# seront eliminees (voir A_2NC).
# Ces lignes doivent etre connectees a au moins 2 autres lignes a chacune des extremités.
# La valeur "0.0" signifie aucun traitement.
#
# A_2NC = Angle de deflexion minimal requis pour eliminer les lignes 2NC.
# Cet angle est forme par le prolongement du dernier segment avec les autres
# lignes connectees aux extremités. Noter que plus l'angle est grand, moins nombreuses
# seront les lignes eliminees.
#
# ILOT = Longueur maximale des lignes a eliminer formant un ilot.
# Tous les ilots dont la longueur est inferieure a la longueur specifiee seront elimines.
# Un ilot est forme par une ligne refermee sur elle-meme.
# La valeur "0.0" signifie aucun traitement.
#
# ILOT_C = Longueur maximale des lignes a eliminer formant un ilot connecte a une ou plusieurs lignes.
# Tous les ilots connectes dont la longueur est inferieure a la longueur specifiee seront
# elimines.
# La valeur "0.0" signifie aucun traitement.
#
=====
# OPTIONS DE LISSAGE, PHASE2 de la AML "gen_mtg".
#
# LIS = Pourcentage de vertex conserves lors du lissage d'une ligne.
# (Valeur comprise entre 0.001 et 1.0)
# La valeur "1.0" signifie aucun traitement.
#
# N_V = Indiquer par O (oui) ou N (non) si le traitement des noeuds virtuels doit etre active.
#
# L_V = Longueur minimale d'un des deux segments d'un vertex pour considerer un noeud virtuel.
# Plus L_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
# A_V = Angle interieur maximal forme par les 2 segments d'un vertex afin de determiner
# un noeud virtuel.
# Plus A_V est grand, plus le nombre de noeuds virtuels crees sera grand.
#
# DI_V = Distance minimale a considerer lors de la creation de noeuds virtuels.
# Pour un vertex donne satisfaisant aux conditions de L_V et A_V, et qu'un noeud virtuel
# existe a l'interieur de ce rayon de recherche sur une meme ligne, aucun noeud virtuel
# ne sera cree.
# Plus DI_V est grand, plus le nombre de noeuds virtuels crees sera faible.
#
=====
# Echelle_entree Echelle_sortie
Echelles= 50000 500000
# Unite de resolution
Resolution= M
#
=====
#TYPE INDICATIF INDICATIF PN RECL DP 2NP 1NP 2NC A_2NC ILOT ILOT_C LIS N_V L_V A_V DI_V
# ENTREE SORTIE
#
L 935 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
# L 955 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 5.0 120.0 5.0
L 956 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 998 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 999 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 1004 @ O 0.0 0.1 7.0 3.0 5.0 45.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 1005 @ O 0.0 0.1 7.0 3.0 5.0 45.0 15.0 10.0 0.85 O 2.0 120.0 2.0
# L 1006 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 5.0 120.0 5.0
# L 1011 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 5.0 120.0 5.0
# L 1012 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 5.0 120.0 5.0
L 1015 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 1016 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 1020 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0
L 1022 @ O 0.0 0.1 7.0 3.0 0.0 25.0 15.0 10.0 0.85 O 2.0 120.0 2.0

```