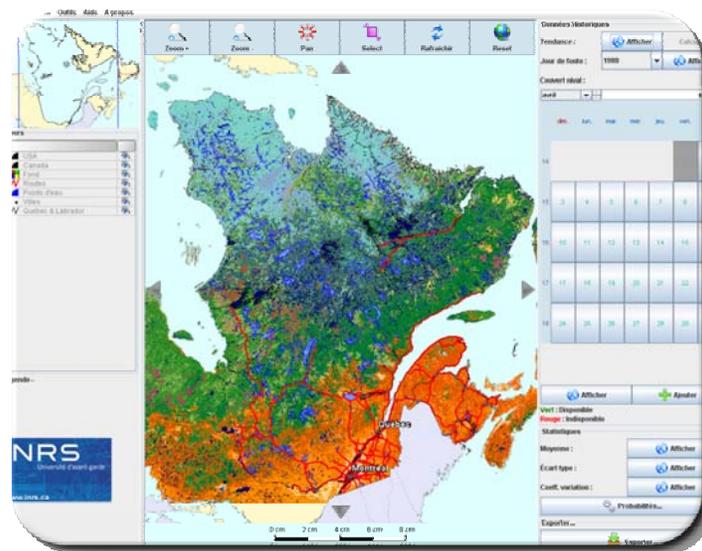


# Développement d'un prototype d'Atlas Interactif du Couvert Nival pour l'Est du Canada

Karem Chokmani, professeur

Benjamin Raphaël-Amanrich, stagiaire

Monique Bernier, professeure



Rapport no. R-1025

Octobre 2008

*ISBN : 978-2-89146-586-1*

# Avant-propos

Le prototype présenté dans ce rapport a été développé par M. Benjamin RAPHAËL-AMANRICH, étudiant en licence et Master 1ère année à l'IUP (Systèmes intelligents) de l'université Paul Sabatier (Toulouse, France), dans le cadre d'un stage effectué à l'INRS-ETE sous la direction de Karem Chokmani, du 03 mars au 08 août 2008.

Le corps du rapport est consacré à présenter la problématique de recherche, la méthodologie de travail ainsi qu'un aperçu de l'Atlas. Des sections les plus importantes du code sont présentées en annexe. Également, une copie sur DVD du prototype accompagne le présent rapport (voir la licence d'utilisation à la page suivante).



# Licence d'utilisation

La version de l'Atlas du couvert nival qui vous est fournie en annexe du présent rapport est encore au stade de développement et n'est toujours pas disponible dans le domaine public. Cette copie de l'Atlas est fournie dans le but d'être testée seulement et sous la condition qu'elle ne soit pas utilisée à des fins pratiques ou de recherche.

Les développeurs de l'Atlas gardent le droit exclusif de publier des informations à partir de cette version de l'Atlas du couvert nival. Aucune présentation, aucun article ou rapport d'aucune sorte utilisant l'Atlas ne peut être publié sans l'accord express de Karem Chokmani. De plus, l'Atlas ne peut être redistribué ni mis à disposition à une tierce partie, sous aucune raison.

*Atlas du couvert nival © 2008, K. Chokmani et al., Institut national de la recherche scientifique, Centre - Eau, Terre, Environnement,*



# Table des matières

1	Introduction.....	1
1.1	Problématique de la recherche .....	1
1.2	Objectifs .....	2
2	Méthodes et matériels .....	5
2.1	Base de données .....	5
2.2	Analyse de la tendance.....	7
2.3	Analyse fréquentielle .....	7
2.3.1	Les Generalized Extrem Values .....	8
3	Implantation du prototype de l'Atlas.....	11
3.1	Plateforme et choix technologiques.....	11
3.1.1	Format des données.....	11
3.1.2	Langage de développement.....	11
3.2	Tour guidé du prototype .....	12
3.2.1	La vue principale .....	13
3.2.1.1	La carte.....	13
3.2.1.2	Les contrôles de navigation .....	14
3.2.1.3	L'échelle .....	16

3.2.2	Les outils Atlas - outils informationnels .....	16
3.2.2.1	La mini-carte .....	16
3.2.2.2	La liste de couches .....	17
3.2.2.3	La légende .....	21
3.2.3	Les outils de données .....	22
3.2.3.1	Les données historiques.....	22
3.2.3.2	La tendance.....	24
3.2.3.3	Les statistiques de bases.....	25
3.2.3.4	L'analyse fréquentielle .....	25
3.2.3.5	L'exportation .....	27
4	Développements futurs .....	31
5	Références .....	35
A.1	Le coin du développeur.....	39
A.2	Exemples de code Java.....	40
A.2.1	<i>La vue de la zone d'exportation</i> .....	40
A.2.2.	Le chargement d'une couche.....	43
A.2.2.1.	Chargement d'un contexte.....	44
A.2.2.2.	Création de la couche à partir du contexte .....	50
A.2.2.3.	Encapsulation d'une couche GeoTools .....	51

A.3	L'Outil Converter .....	52
A.4	DVD du prototype de l'Atlas .....	53



# Liste des figures

Figure 1 - Région d'étude (Est du Canada).....	6
Figure 2 - Exemple d'ajustement d'une GEV (bleu) à une variable aléatoire asymétrique (histogramme gris).....	9
Figure 3 - Interface usager de l'Atlas.....	12
Figure 4 - Exemple du menu déroulant .....	13
Figure 5 - Échelle de la carte principale.....	16
Figure 6 - Mini-carte .....	17
Figure 7 - Liste des couches avec les couches par défauts (Cosmétique et Occupation du sol) et une couche usager (Jour de fonte en 1988).....	18
Figure 8 - Exemple du menu contextuel. Ici la transparence est défini à environ 50%. .....	18
Figure 9 - Exemple représentatif d'une approche par couches .....	20
Figure 10 - Boite de dialogue permettant la modification de la légende.....	22
Figure 11 - Calendrier de sélection du jour pour l'affichage du Couvert Nival (ici positionné sur le moi d'avril 1988) .....	24
Figure 12 - Probabilité de fonte au 110 <sup>ième</sup> jour de l'année dans la région de la ville de Québec.....	26
Figure 13 - Jour de fonte probable à 50% dans la région autour de la ville de Québec .....	27
Figure 14 - Capture d'écran de la boite de dialogue d'exportation .....	28
Figure 15 - Schéma de conception .....	39
Figure 16 - Capture écran de la sortie de l'outil <i>Convertir</i> .....	52



# 1 INTRODUCTION

---

## 1.1 Problématique de la recherche

De nombreux travaux ont été consacrés à l'étude de la variabilité temporelle du couvert nival au Canada, à travers l'analyse des séries temporelles de cartes de neige issues des données de télédétection ou de la régionalisation des observations locales (Brown et Braaten, 1998; Brown *et al.*, 2003). Une grande part de ces travaux a été dédiée à l'étude de la variabilité en lien avec les changements climatiques (Brown et Goodison, 1996; Derksen *et al.*, 2002; Rikiishi *et al.*, 2004; Kitaev *et al.*, 2005; Latifovic *et al.*, 2005). Ces travaux ont révélé l'existence, à l'échelle de l'hémisphère Nord, d'une tendance à la baisse dans la durée et dans l'étendue du couvert nival. Cependant, peu d'études ont été réalisées au niveau régional, en particulier l'Est du Canada et dans lesquelles il a été question non seulement de la variabilité temporelle (interannuelle) mais aussi de la variabilité spatiale en lien avec les changements climatiques (au niveau du patron spatial de l'enneigement). Par ailleurs, l'étude de la variabilité temporelle s'est généralement limitée à la détection de la tendance dans les valeurs centrales de la distribution (moyenne, médiane) ou les valeurs extrêmes au détriment de la variance. D'autre part, les séries temporelles de données sur la neige ont servi à bâtir des atlas de neige (CRYSYS, 2007; Ressources Naturelles Canada, 2007) qui se limitent aux valeurs centrales de la variabilité des paramètres descripteurs de la neige (p.ex. : la valeur maximale ou minimale de l'équivalent en eau de la neige, la profondeur maximale de la neige, la durée du couvert nival, la date de l'apparition ou de la disparition du couvert, etc.). Aucune information n'est

disponible sur la nature de la distribution statistique de ces paramètres et encore moins sur la probabilité de leur occurrence. Par ailleurs, des atlas régionaux sur la durée et la qualité de l'enneigement sont quasi inexistant.

Après une recherche rapide sur le web, l'Atlas du Canada sur la durée de l'enneigement disponible en ligne se contente de donner la date moyenne d'établissement et de disparition du couvert nival ainsi que la durée moyenne (<http://www.socc.ca/snow/atlas/index.cfm>).

## 1.2 Objectifs

Développer un prototype d'un Atlas numérique et interactif de la fonte de la neige en se basant sur les cartes historiques de l'étendue du couvert nival pour l'Est du Canada produites par Chokmani et al. (2007) et ce, pour la période de fonte d'avril à mai de 1988 jusqu'à 1999.

L'Atlas se doit être :

- 1) convivial : facile d'utilisation pour des non-initiés aux systèmes de gestion des données spatiales;
- 2) portable : indépendant de la plateforme utilisée;
- 3) ouvert : pouvant être enrichi par des nouvelles données et permettant l'exportation des données qui y sont stockées; et
- 4) évolutif : permettant l'ajout de données supplémentaires en élargissant l'étendue de la série temporelle, l'ajout de nouveaux paramètres du

couvert nival tel que l'équivalent en eau de la neige, ou l'élargissement du territoire à l'étude.

Par définition, un atlas c'est un « recueil ordonné de cartes, conçu pour représenter un espace donné et exposer un ou plusieurs thèmes <sup>1</sup>». En l'occurrence ici, le thème représenté est l'état du couvert nival en période de fonte.

Le prototype d'atlas doit permettre d'explorer, manipuler et exporter les données historiques déjà disponibles dans la base de données. À partir de ces données, le prototype doit également permettre de générer sous forme cartographique non seulement les valeurs centrales et la dispersion des paramètres physiques du couvert nival en question (moyenne, médiane, mode, écarts types, coefficient de variation, etc.) mais aussi d'en définir les probabilités d'apparition d'un événement d'une valeur donnée ou la valeur d'un événement avec un niveau de probabilité donné. L'Atlas doit aussi permettre de vérifier la présence éventuelle d'une tendance dans les données historiques utilisées. Les fonctionnalités de visualisation et d'exportation développés pour les données historiques doivent être également applicables aux paramètres statistiques.

---

<sup>1</sup> Le Grand Dictionnaire terminologique. Office québécois de la langue française : [http://www.granddictionnaire.com/btml/fra/r\\_motclef/index800\\_1.asp](http://www.granddictionnaire.com/btml/fra/r_motclef/index800_1.asp)



## 2 MÉTHODES ET MATÉRIELS

---

### 2.1 Base de données

La base de données cartographique soutenant l'atlas consiste en une cartographie historique de la présence de la neige effectuée à l'aide d'un algorithme de fusion de données satellitaires développé par Chokmani et al. (2007). Les données issues de cet algorithme couvrent la période de fonte c.-à-d. du 1er avril au 31 mai de chaque année, de 1988 à 1999. La précision globale de l'algorithme est estimée à 86% (en comparaison avec les observations de la hauteur de neige au niveau des stations météorologiques).

L'algorithme de fusion a été développé dans le cadre d'un projet visant la validation de la composante neige du Modèle Régional Canadien du Climat - MRCC (projet CRSNG/RDC financé par Hydro-Québec et le Consortium OURANOS). L'algorithme consiste à combiner les résultats de cartographie de la neige en utilisant les données dans le visible et l'infrarouge du capteur NOAA-AVHRR (dont la résolution spatiale est de 1 km et dont l'utilisation est limitée par la présence des nuages) avec ceux de la cartographie de la neige réalisée à l'aide du capteur micro-ondes passives SSMI (qui est insensible à la présence des nuages mais sa résolution spatiale est de 25 km). Ceci tire avantage des caractéristiques des deux capteurs tout en minimisant les limites à leur application et produire ainsi une série de cartes quotidiennes de la neige durant les mois d'avril et mai de chaque année des 12 années ciblées par l'étude (1988-1999). Les cartes ainsi produites sont exemptes de nuages et ont une résolution nominale de 1 km x 1 km.

La région couverte par les données englobe l'Est du Canada : la Québec, le Labrador, le Nouveau Brunswick et la Nouvelle-Ecosse (Figure 1).

À partir des cartes journalières du couvert nival, une cartographie du jour de fonte a été produite pour chacune des 12 années ciblées par l'étude (1988-1999). Pour une année donnée et pour chaque pixel donné de la carte, le jour de fonte correspond au jour de la fin de la période de fonte et a été défini comme étant le premier jour sans neige se situant immédiatement après le dernier jour de présence de neige de la saison détectée par l'algorithme.

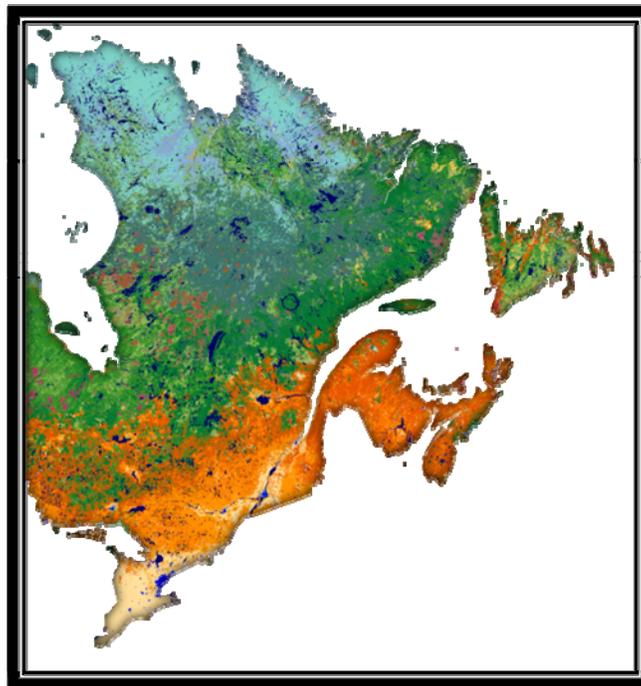


Figure 1 - Région d'étude (Est du Canada)

## 2.2 Analyse de la tendance

Afin de détecter la présence éventuelle d'une tendance dans les données de jour de fonte, le test non paramétrique de Mann-Kendall a été implanté dans l'Atlas (ÖNÖZ et Bayazit, 2003). Ce test est basé sur la statistique  $S$  qui est calculée comme suit :

$$S = P - M \quad (1)$$

Où  $P$  est le nombre de paires d'observations  $y_i, y_j$  ( $i > j$ ,  $i$  et  $j$  indiquent l'année) pour lesquelles  $y_i > y_j$  et  $M$  est le nombre de paires d'observations  $y_i, y_j$  ( $i > j$ ,  $i$  et  $j$  indiquent l'année) pour lesquelles  $y_i < y_j$ .

Pour  $n > 10$ , où  $n$  est la taille de l'échantillon (dans notre cas  $n=12$ ), la distribution de  $S$  est comme suit.

$$Z = \begin{cases} \frac{S-1}{\sigma_s} & \text{si } s > 0 \\ 0 & \text{si } s = 0 \\ \frac{S+1}{\sigma_s} & \text{si } s < 0 \end{cases} \quad (2)$$

Où  $Z$  suit la distribution normale standard et  $\sigma_s = \sqrt{\frac{n(n-1)(2n+5)}{18}}$

L'hypothèse nulle qu'il n'y a pas de tendance est refusée lorsque la valeur  $Z$  calculée est supérieure à  $Z_{\alpha/2}$ , en valeur absolue.

## 2.3 Analyse fréquentielle

En se basant sur la série temporelle de cartes de jour de fonte, une approche d'analyse fréquentielle de ce paramètre a été implantée dans l'Atlas. L'analyse fréquentielle est une méthode statistique de prédiction consistant à étudier des données historiques, caractéristiques d'un processus donné (état du couvert nival en

l'occurrence le jour de fonte), afin d'en définir les probabilités d'apparition d'un événement de valeur donnée. L'analyse fréquentielle comprend outre la constitution de la série historique de données (dans notre cas nous disposons d'une série de 12 cartes pour la période 1988-1999), le choix et l'ajustement du modèle fréquentielle (fonction de distribution des probabilités). Ainsi, il sera par exemple possible de calculer la valeur du jour de fonte, à un point donné du territoire en fonction sa probabilité d'occurrence. Ceci représente la valeur ajoutée principale de l'Atlas. Par ailleurs, l'Atlas peu fourni les statistiques de bases sur le jour de fonte. Ces calculs ont été réalisé offline<sup>2</sup> par le biais du progiciel Matlab<sup>®</sup>.

### *2.3.1 Les Generalized Extrem Values*

Pour le choix du modèle fréquentielle (fonction de distribution des probabilités), nous avons opté pour la Generalized Extrem Value - GEV (El Adlouni *et al.*, 2007; Wikipédia, 2008). La GEV est une famille de distributions de probabilité continues, flexible et adaptée à des petites séries de données comme la notre. Elle a été conçue pour décrire la fonction de distribution des probabilités des valeurs extrêmes d'une variable aléatoire (minimum, maximum, premier, dernier, etc.) qui sont généralement des fonctions fortement asymétriques, ce qui la GEV d'autant plus adaptée aux données de jours de fonte. La Figure 2 illustre bien la capacité d'un GEV à s'adapter à un jeu de données dont la distribution est asymétrique.

---

<sup>2</sup> Un calcul offline est un calcul qui ne peu être réalisé par l'Atlas de la fonte de la neige, et nécessite donc l'utilisation d'autres outils.

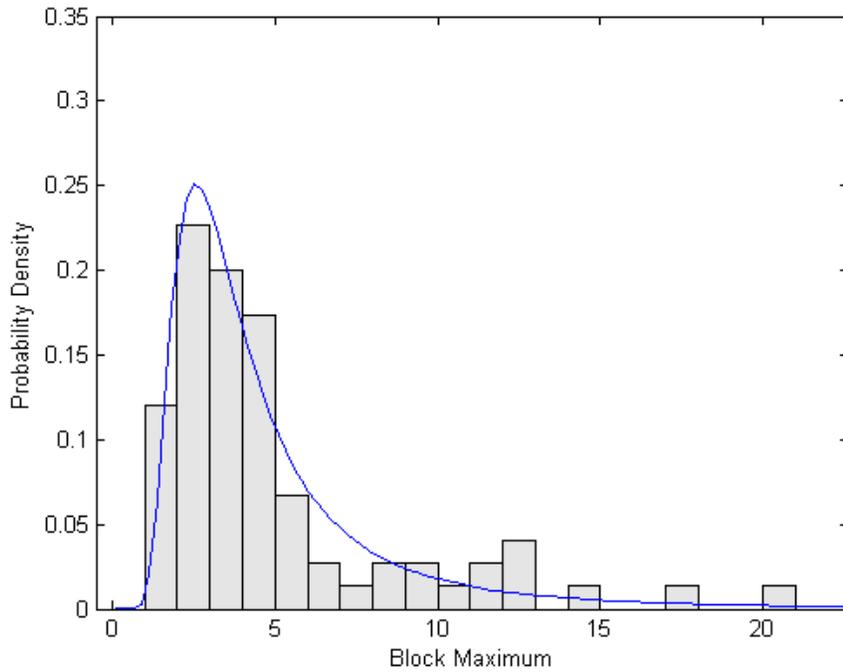


Figure 2 - Exemple d'ajustement d'une GEV (bleu) à une variable aléatoire asymétrique (histogramme gris)

La GEV possède trois paramètres :

- $\kappa$  : Kappa, le paramètre de forme
- $\mu$  : Mu, le paramètre de localisation
- $\alpha$  : Alpha, le paramètre d'échelle

Ces paramètres permettent de calculer la fonction cumulative de distribution des probabilités comme suit :

$$\begin{aligned}
 F_{GEV}(x) &= \exp\left[-\left(1 - \frac{\kappa}{\alpha}(x - \mu)\right)^{\frac{1}{\kappa}}\right] & \kappa \neq 0 \\
 &= \exp\left[-\exp\left(-\frac{(x - \mu)}{\alpha}\right)\right] & \kappa = 0
 \end{aligned}
 \tag{3}$$

Où  $x$  représente la variable aléatoire, en l'occurrence le jour de fonte. Ainsi, la fonction de cumulative donnée par l'équation 3 permet d'obtenir pour chaque pixel, la probabilité d'obtenir un jour de fonte égale ou inférieur à la valeur donnée  $x$ . En calculant la fonction inverse de la fonction cumulative, il est possible d'obtenir la valeur du jour de fonte correspondant à un niveau de probabilité donné.

Le calcul des paramètres de la GEV a été effectué offline dans le progiciel Matlab<sup>®</sup> à l'aide de l'approche de maximum de vraisemblance.

## 3 IMPLANTATION DU PROTOTYPE DE L'ATLAS

---

### 3.1 Plateforme et choix technologiques

#### *3.1.1 Format des données*

Tous les éléments de la base de données images implantée dans l'Atlas : données historiques brutes (cartes journaliers du couvert nival et cartes de jour de fonte) ainsi que les couches de paramètres statistiques calculés ont été encodées en format GeoTiff. C'est un format très largement utilisé pour l'échange de données matricielles (images) entre logiciels de gestion de données à référence spatiale. Ce format offre un bon niveau de compression tout en permettant de conserver et de gérer l'information relative au système de coordonnées géographiques de la carte/image en question.

#### *3.1.2 Langage de développement*

Étant donné que l'Atlas se doit d'être portable, le langage de développement retenu a été le Java (Version 6, mise à jour 7, build 1.6.0\_07-b06). En effet, ce langage est souvent synonyme de portabilité. Un programme développé dans ce langage est fonctionnel peu importe le système d'exploitation installé sur la machine. Il suffit d'installer au préalable la dernière version de Java sur son poste. De plus, ce langage est connu pour être très efficace quant à la création et la manipulation des interfaces graphiques.

Les fonctionnalités d'édition et de gestion des données à référence spatiale (zoom, sélection spatiale, édition des couches cartographiques, exportation des données images, etc.) ont été développées à l'aide de la librairie du domaine libre GeoTools

(GeoTools, 2008). Celle-ci permet de développer des outils de gestion de l'information géographique selon les normes de l'Open Geospatial Consortium (OGC, 2008). Par ailleurs, la librairie GeoTools est complètement supportée par Java

### 3.2 Tour guidé du prototype

L'interface usager de l'Atlas a été subdivisée en trois sections distinctes (Figure 3) : la vue principale (zone d'affichage des cartes), la section informationnelle sur les cartes (Outils Atlas), et la section spécifique à la sélection des données à afficher (Outils données).

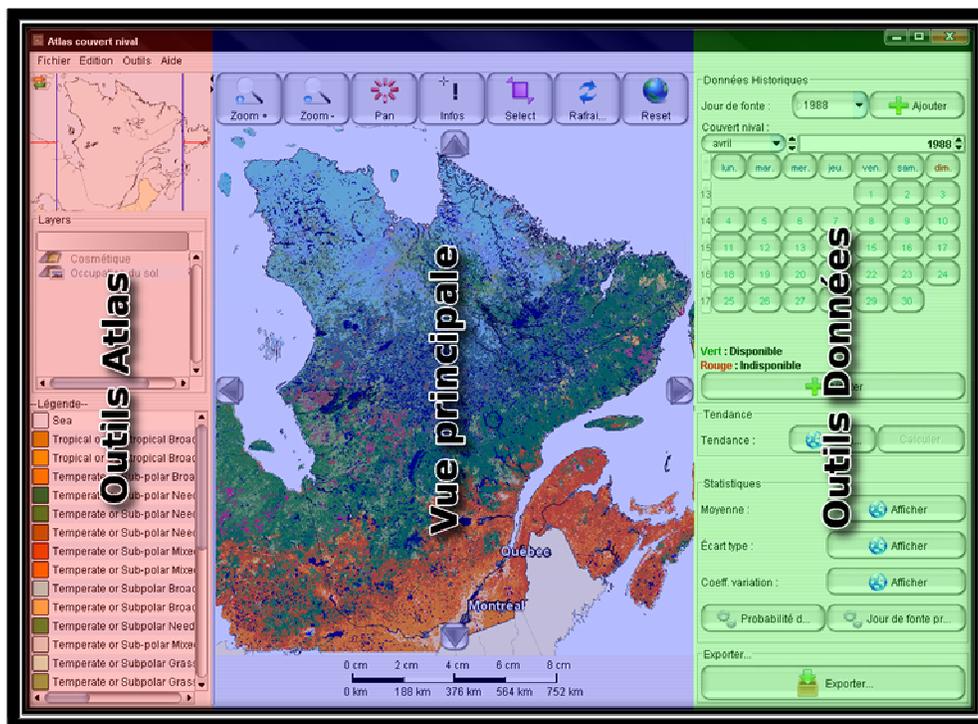


Figure 3 - Interface usager de l'Atlas

Toutes les fonctionnalités de l'Atlas sont également accessibles via les menus déroulants de l'application, en haut à gauche de l'interface (Figure 4). Cela permet une navigation sans l'utilisation de la souris. Il manque encore les touches de raccourcis permettant d'accéder à toutes ces fonctionnalités plus rapidement.

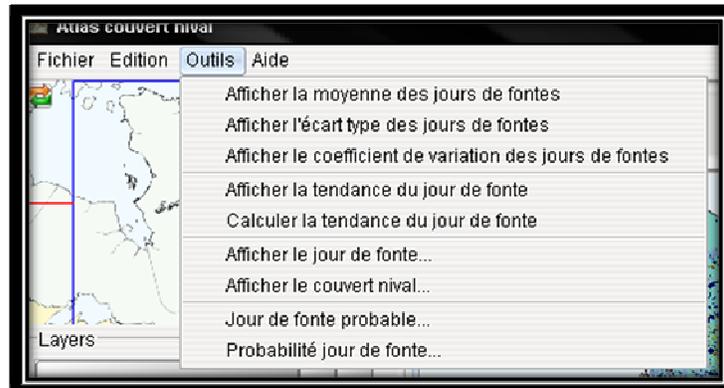


Figure 4 - Exemple du menu déroulant

### 3.2.1 La vue principale

La zone d'affichage principale des cartes est composée de la carte, des différents contrôles de navigations et de l'échelle.

#### 3.2.1.1 La carte

C'est la composante graphique principale qui permet d'afficher les différentes couches de données sous forme cartographique. Cette fonctionnalité est gérée par la librairie GeoTools. Ainsi seule la manipulation des données est laissée au programmeur. Les cartes affichées peuvent être de deux types :

### *Vecteurs*

Représentés sous forme de lignes, ou de points. Ils servent à représenter des éléments géographiques tels que les frontières administratives, les routes principales ou les principales villes agglomérations. Ce sont des données spatiales auxiliaires qui servent à aider l'utilisateur à se situer dans l'espace.

### *Raster*

L'ensemble des données sur le couvert nival (données historiques, paramètres statistiques) aussi bien que l'information sur l'occupation du sol sont encodées en format « raster » ou format matriciel. En fait, ce sont des images constitués de matrices de points (pixels). Chaque pixel est géo-référencé, c'est à dire qu'il possède une coordonnée unique selon un système de projection défini par le système.

#### *3.2.1.2 Les contrôles de navigation*

Ceux-ci permettent à l'utilisateur de naviguer au sein de la carte, et sont directement liés aux actions possibles sur la carte. Ainsi les contrôles proposés sont :

#### *Le zoom*

Il est possible de faire des zooms arrière et des zooms avant. Ceci permet à l'utilisateur soit d'avoir une vue globale de la carte ou d'examiner en détail des secteurs particuliers de la carte. À noter que l'outil de zoom est directement connecté aux données du système afin de réguler l'affichage des noms des villes. En effet, plus on est proche plus il sera nécessaire d'avoir le détail des petites villes, et inversement, lorsque le zoom est minimal, seuls les noms des

agglomérations les plus importantes telles que Montréal et Québec sont affichés.

### *Le pan*

À un niveau zoom donné, cet outil permet de déplacer la carte d'un clic maintenu de la souris.

### *L'information*

En cliquant sur ce bouton l'utilisateur peut se mettre en mode « information », et ainsi connaître la nature de l'information actuellement affichée. Ainsi en cliquant sur la carte, l'information contenue dans le pixel sous la zone de clic est affichée. Cette information dépend donc de la couche actuellement affichée ou sélectionnée dans la liste de couches (présenté plus en avant, p17 : 3.2.2.2. La liste de couches).

### *La sélection*

En appuyant sur ce bouton, l'utilisateur peut effectuer la sélection d'une région, qui apparaîtra en surbrillance verte, ou bien, dans le cas où une sélection serait déjà présente, supprimer la sélection déjà présente. Il ne peut y avoir qu'une seule sélection en cours, donc suite à une première sélection, la seconde écrase la première.

### *Le rafraîchissement et la réinitialisation*

L'utilisateur peut donc demander le rendu de la carte dans le cas d'un artefact d'affichage, ou bien il peut réinitialiser les paramètres de positionnement et de zoom de la carte.

A tous ces boutons de navigation s'ajoutent des flèches directement présentent sur la carte, permettant de déplacer la carte vers les quatre directions cardinales.

### 3.2.1.3 L'échelle

Afin de donner à l'utilisateur un repère sur les dimensions réelles des secteurs affichés, une échelle (cf. Figure 5) double (distance en cm sur l'écran et celle correspondante sur le terrain en km) est incorporée à la vue principale. Celle-ci s'ajuste automatiquement selon le niveau de zoom. Le développement de cette échelle a été effectué à l'aide des normes Swing (normes de composant graphiques Java) et fait appel à des notions avancées de développement orienté objet appelées *Design Pattern MVC*. C'est une notion (voir Annexes)



Figure 5 - Échelle de la carte principale

### 3.2.2 Les outils Atlas - outils informationnels

Ces outils permettent d'afficher des informations, ou encore de manipuler des couches d'informations. Ainsi, trois sections sont présentes dans cette section : la mini-carte, la liste des couches, et la légende.

#### 3.2.2.1 La mini-carte

La mini-carte permet à l'utilisateur de se localiser sur la carte principale. En effet, lorsque l'utilisateur effectue un zoom, et qu'il se déplace, il lui est très utile d'avoir la

possibilité de voir où est située la vue active. Ainsi, la vue principale est représentée par un carré bleu sur la mini-carte (Figure 6). Des guides sous formes de lignes rouges fournissent une aide supplémentaire.

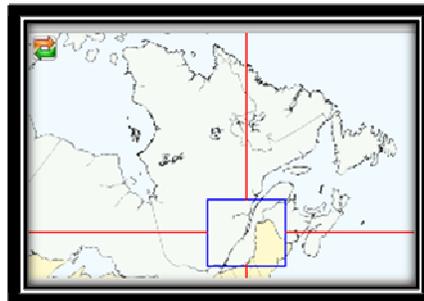


Figure 6 - Mini-carte

L'utilisateur a la possibilité de naviguer directement dans la mini-carte. Ainsi, lorsqu'il clique sur celle-ci, la vue principale va immédiatement se recentrer sur le clic. Il existe donc un lien entre ce que voit l'utilisateur et ce qui est affiché sur la mini-carte. D'un point de vue technique, la mini-carte utilise le même principe que la carte principale, mais affiche seulement certaines couches, telles que les frontières administratives, afin d'avoir un rendu moins chargé donc plus simple d'accès pour l'utilisateur.

#### 3.2.2.2 La liste de couches

Cet outil permet d'afficher la liste des couches actuellement présentes sur la carte principale. Il y est possible de manipuler le statut de ces couches. Les icônes représentant des yeux (👁️ ou 👁️) affichées devant chaque nom de la liste des couches permettent d'un seul clic de la part de l'utilisateur de rendre la couche correspondante visible ou invisible (Figure 7). Le point d'exclamation (!) signifie que c'est de

cette couche que l'utilisateur va extraire les informations avec le contrôle d'information (cf. p14, 3.2.1.2. Les contrôles de navigation).



Figure 7 - Liste des couches avec les couches par défauts (Cosmétique et Occupation du sol) et une couche usager (Jour de fonte en 1988)

En cliquant avec le bouton droit de la souris sur l'une des couches de la liste, un menu contextuel s'affiche permettant en plus de supprimer la couche de la liste (et par conséquent de la vue principale) ou de modifier son nom de régler le niveau de transparence de celle-ci (Figure 8).



Figure 8 - Exemple du menu contextuel. Ici la transparence est définie à environ 50%.

Il nécessaire de préciser qu'il existe deux types de couches distincts :

Les couches systèmes :

Celles-ci sont créées au démarrage du système, et ne peuvent être supprimées par l'utilisateur. Ce sont des couches qui permettent à l'utilisateur de mieux appréhender le travail sur lequel il travaille. On y retrouve donc ce que l'on appelle la couche *Cosmétique*, composée des frontières administratives et politiques et des villes, et la couche *Occupation du sol*, qui permet à l'utilisateur de visualiser les différents types de couverture du sol présents sur le territoire.

Toutefois, plusieurs difficultés sont afférentes à l'affichage de la couche *Cosmétique*. En effet, la librairie *GeoTools* gère le rendu de la carte principale comme un empilement successif de ces couches. Ainsi la dernière couche activée est affichée par dessus toutes les couches précédentes. Or par un souci de cohérence, la couche *Cosmétique* est composée en fait de plusieurs couches - cinq en fait - dont certaines se doivent d'être toujours en arrière plan, par exemple les frontières des États-Unis, et d'autres toujours en avant plan, tel que les villes ou encore la frontière du Québec et du Labrador. La Figure 9 représente un exemple d'une structure par couches. Ici la couche *Cosmétiques* serait en A (on aurait en 1 le fond, comme par exemple les frontières du Canada et des États-Unis, et en 5 les villes et la frontière du Québec et Labrador), la couche *Occupation du sol* serait en C, et enfin les couches usager en B.

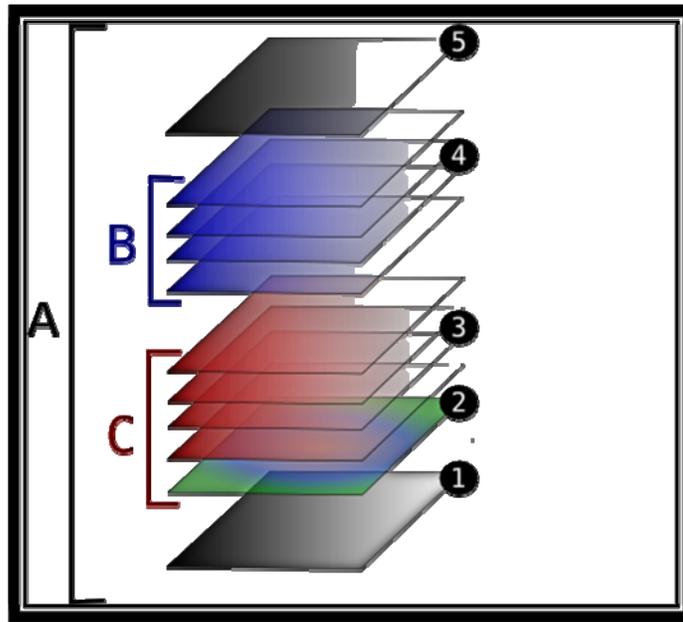


Figure 9 - Exemple représentatif d'une approche par couches

D'un point de vue technique, lorsque l'utilisateur déplace une couche par un *Drag And Drop*<sup>3</sup> dans la liste de couches, et ce afin de modifier l'ordre d'affichage de ces couches, le système doit toujours vérifier à correctement replacer les couches internes de la couche *Cosmétiques* (faire en sorte que la partie *Occupation du sol* et les couches ajoutées par l'utilisateur restent bien entre les couches *Cosmétiques*). Ceci est géré par le mécanisme de *Drag And Drop* de Java. Dans le cadre de ce mécanisme, l'information sur la position de la couche dans la liste des couches a été ajoutée. Ainsi, à la place d'avoir une couche classique de la librairie *GeoTools*, l'Atlas manipule des couches qui se souviennent de leur position d'affichage dans la liste de couches, mais qui peuvent aussi :

---

<sup>3</sup> Le glisser-déposer (de l'anglais *drag-and-drop*) est dans un environnement graphique une méthode consistant à utiliser un dispositif de pointage (généralement une souris) pour déplacer d'un endroit à un autre un élément graphique présent sur l'écran d'un micro ordinateur (Wikipédia, Glisser-déposer, 2008).

- Formater le texte affiché lorsque l'utilisateur demande de l'information selon le contenu de la couche (cf. p14, 3.2.1.2. Les contrôles de navigation)
- Qui contiennent la légende (cf. p21, 3.2.2.3. La légende) associée à cette couche.

Pour des détails sur la création de couches étendues, veuillez vous référer à l'Annexe (page A. 51 : A.2.2.3. Encapsulation d'une couche GeoTools).

### Les couches usagers :

Ce sont les couches ajoutées par l'utilisateur lors de l'utilisation de l'Atlas. Elles se distinguent des couches systèmes au sens où elles n'ont pas la même icône et peuvent être supprimable en appuyant sur la touche « SUPPR », ou dans le menu contextuel. Bien sûr, toutes les couches se voient en mesure d'être rendues visible ou transparente, on peut aussi modifier le nom de chacune d'entre elles.

#### *3.2.2.3 La légende*

Une autre des composantes essentielles de la section informationnelle est la légende. Cet encart illustre les symboles et les codes de couleurs employés pour afficher chacune des couches dans la vue principale.

L'affichage de chacune des couches est dépendant du *ColorMap*. Celui-ci définit quelle est la couleur affectée à une certaine gamme de valeurs de la variable ou paramètre affiché. Ainsi, pour chacun des types d'informations contenus dans les couches usager, un *ColorMap* différent lui est associé.

L'utilisateur a la possibilité, par le biais du composant de la légende, de modifier le *ColorMap* et d'éditer le nom de chacun des symboles (Figure 10). Il peut donc attribuer un autre nom à un symbole et une autre couleur.

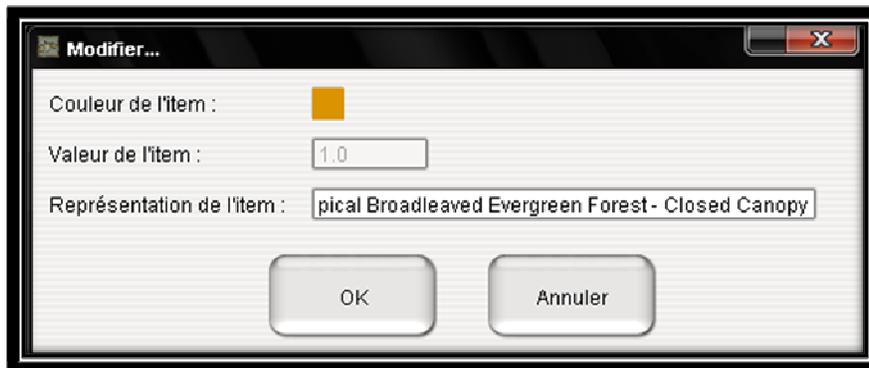


Figure 10 - Boite de dialogue permettant la modification de la légende

### 3.2.3 Les outils de données

Le but de l'Atlas est de rendre disponible et ce, d'une manière interactive, les données historiques sur le couvert nival en période de fonte (avril-mai) et pour la période 1988-1999 ainsi qu'à les paramètres statistiques dérivés à partir de ces données.

#### 3.2.3.1 Les données historiques

Il est intéressant de noter que l'Atlas a été conçu de manière à pouvoir ajouter d'éventuelles données supplémentaires dans la base de données, celles-ci deviendront alors directement exploitable par l'Atlas, et donc l'utilisateur.

À noter aussi que dans un souci de cohérence et d'allègement de la mémoire, les couches déjà chargées en mémoire dans le système ne peuvent s'afficher en double, et seront simplement rapportées en avant plan si l'utilisateur demande le réaffichage.

Pour consulter le code source permettant de charger des données dans l'Atlas, veuillez vous référer à l'Annexe.

### *Le dernier jour de fonte*

La première partie concerne le dernier jour de fonte des années 1988 à 1999. L'utilisateur choisit l'année dont il désire connaître le dernier jour de fonte et peut afficher la couche dans la vue principale en cliquant sur le bouton « Ajouter » vis-à-vis la zone d'affichage de l'année.

L'information est chargée l'image correspondante en format GeoTiff et construire une couche en utilisant les outils fournis par la librairie GeoTools. Lors de l'ajout la couche dans le contexte général de la carte principale, un mécanisme automatique, basé sur un *Design Pattern* appelé *Design Pattern Observer*, observe cet ajout et va construire automatiquement la version étendue de cette couche et mettre à jour la liste des couches et la légende. Tous les mécanismes d'ajouts de données sont basés sur le même principe.

### *L'état du couvert nival à une date donnée*

La deuxième partie concerne les cartes journalières de l'état du couvert nival produites par l'algorithme de fusion NOAA-SSM/I. Celles-ci couvrent 60 jours par années (1988-1999), ce qui représente un total de 720 images.

Afin d'offrir un moyen simple pour l'utilisateur de se retrouver dans tant de données, un calendrier interactif a été implanté (Figure 11). Celui-ci permet la navigation aisée dans les dates, et affiche les jours pour lesquels des données sont disponibles dans la base de données en vert et les autres jours en rouge. L'utilisateur n'a donc qu'à

sélectionner le jour qu'il souhaite consulter et demander son affichage. D'un point de vue critique il serait sûrement intéressant de restreindre la navigation dans le temps aux seuls jours disponibles ; actuellement le système n'a aucune restriction temporelle.



Figure 11 - Calendrier de sélection du jour pour l'affichage du Couvert Nival (ici positionné sur le moi d'avril 1988)

### 3.2.3.2 La tendance

Utilisant l'algorithme de détection de tendance décrit plus haut (ÖNÖZ et Bayazit, 2003), l'application est en mesure de calculer et d'afficher, la tendance dans les données du dernier jour de fonte à partir des données disponibles. Ainsi, l'utilisateur peut savoir si dans une région donnée le dernier jour de fonte à tendance à arriver plus tardivement, plus tôt, ou alors qu'il n'y a aucune évolution significative. Pour qu'une évolution ( $Z$ ) soit significative il faut qu'elle reste dans l'intervalle de confiance à 95% ; ce qui veut dire, par exemple, que si la tendance est positive, donc, dans 95% des cas, le dernier jour de fonte à tendance à arriver plus tardivement.

### 3.2.3.3 Les statistiques de bases

L'Atlas peut également fournir à l'utilisateur les statistiques de bases sur les derniers jours de fontes. Dans cette version du prototype, l'Atlas propose le jour moyen de fonte sur les douze années d'étude, l'écart type et le coefficient de variation<sup>4</sup>. Ces calculs ont été réalisés offline par le biais de l'outil Matlab<sup>®</sup>. Les données sont produites en format texte, et converties par un utilitaire dédié, appelé *Converter*, en image GeoTiff. Cet utilitaire développé par nous dans l'environnement Matlab<sup>®</sup> et sont présents sur le DVD de l'Atlas, permettant ainsi à l'utilisateur de recalculer les images GeoTiff en cas de modification des données historiques (ajout ou suppression d'années d'étude). Pour de plus amples informations sur l'outil *Converter*, veuillez vous référer à l'Annexe.

### 3.2.3.4 L'analyse fréquentielle

Des fonctions cumulatives de distribution des probabilités ont été ajustées aux séries temporelles de jour de fonte (voir section page 7, Analyse fréquentielle) permettant de généraliser le comportement de chaque pixel du territoire d'un point de vue probabiliste. Le calcul est effectué offline dans le progiciel Matlab<sup>®</sup>. Ceci a permis d'estimer chacun des trois paramètres pour tous les pixels du territoire à l'étude. Toutes les fonctions Matlab<sup>®</sup> permettant de réaliser ce calcul sont présentes sur le DVD de l'Atlas de la fonte de la neige, et donc disponible à l'utilisateur s'il désire refaire les calculs (en cas de changements dans les données sources par exemple).

---

<sup>4</sup> Le coefficient de variation est une mesure de la dispersion relative, il se calcul comme le rapport entre l'écart-type et la moyenne (Wikipédia, Coefficient de variation, 2008).

*La probabilité d'occurrence de la fonte de la neige pour un jour donné ou avant*

L'utilisateur spécifie le jour dont il désire connaître la probabilité de fonte, et l'Atlas calcule alors cette probabilité à l'aide des paramètres préétablis de la fonction cumulative de distribution des probabilités. Lorsque la probabilité indique 100% alors, c'est qu'à cette date, la neige a 100% de chances d'être fondue. Le résultat de probabilité est calculé pour chacun des pixels de la région d'étude. Dans cet exemple présenté dans la Figure 12, l'utilisateur a demandé la probabilité pour que la fonte ait eu lieu avant le 110<sup>ème</sup> jour julien (le 20 avril), sur la région autour de la ville de Québec. Le vert représente les valeurs proches de 0%, et le rouge proches de 100%. Dans ce cas, la probabilité pour la ville de Québec est supérieure à 94%.

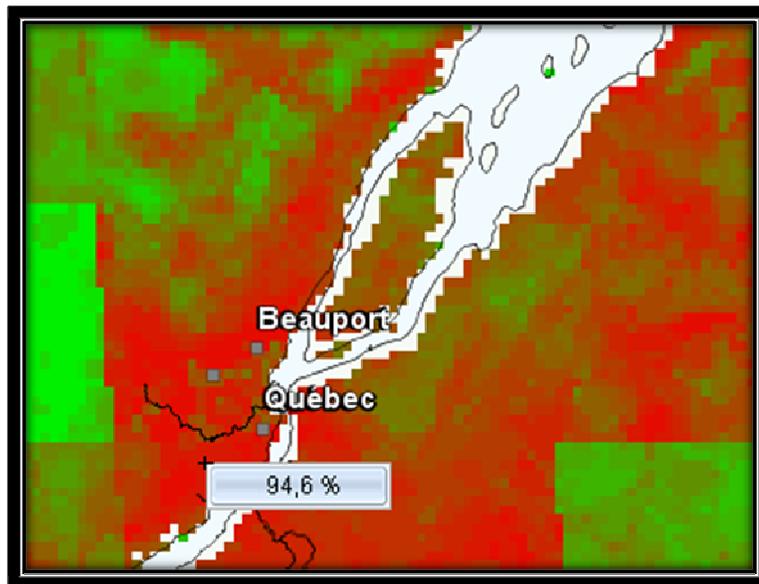


Figure 12 - Probabilité de fonte au 110<sup>ème</sup> jour de l'année dans la région de la ville de Québec

### *La valeur due jour de fonte pour une probabilité donnée*

Ceci est le cheminement inverse, et utilise donc la formule inverse de la fonction cumulative de distribution de la probabilité de chacun des pixels. Ainsi l'utilisateur entre une valeur de la probabilité en pourcentage et l'Atlas lui renvoie une couche contenant le jour de fonte maximal correspondant à cette valeur de probabilité de fonte.

Dans l'exemple illustré par la Figure 13, l'utilisateur a spécifié une probabilité de 50%. Ainsi, pour la région autour de la ville de Québec, la neige a 50% de chance de disparaître le 91<sup>ème</sup> jour julien ou avant (01 avril). La légende va du bleu (91<sup>ème</sup> jour et moins) au rouge (152<sup>ème</sup> jour et plus).

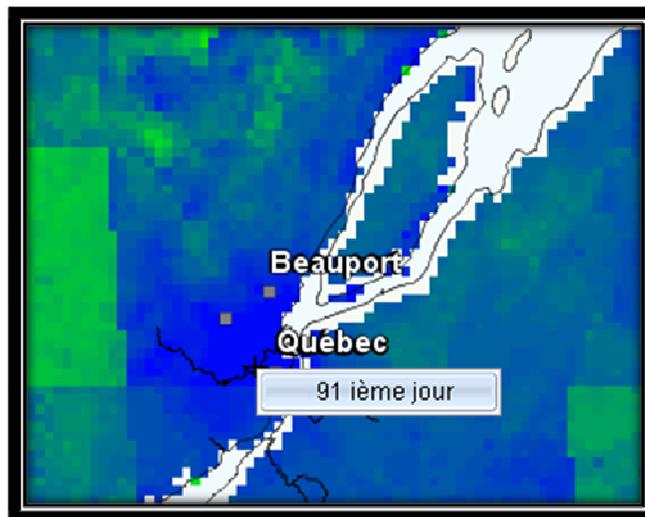


Figure 13 - Jour de fonte probable à 50% dans la région autour de la ville de Québec

#### *3.2.3.5 L'exportation*

Afin de répondre à l'exigence que l'Atlas se doit être ouvert et permettre des échanges de données avec des systèmes externes, un outil d'exportation des données

a été conçu. Cet outil permet d'exporter les couches usagers sous deux formats. Le premier est le format d'origine, (GeoTiff), et le second est un format texte (ASCII). Ces deux formats sont reconnus par la plus part des systèmes de traitement de données (à référence spatiale ou non).

Du point de vue conception, l'outil d'exportation est une fenêtre indépendante connectée aux données de l'application (Figure 14). Elle récupère la sélection - si elle existe - et les couches usagers, et s'occupe de tous les processus d'exportation (voir l'Annexe - A.2.1 *La vue de la zone d'exportation*, pour des détails sur le code utilisé pour l'affichage de la sélection). Entre autre fonctionnalités, cet outil permet de sélectionner les couches à exporter, de modifier la région sélectionnée, de donner un nom au groupe d'export (nom qui apparaîtra au début de tous les fichiers exportés durant la session d'exportation).

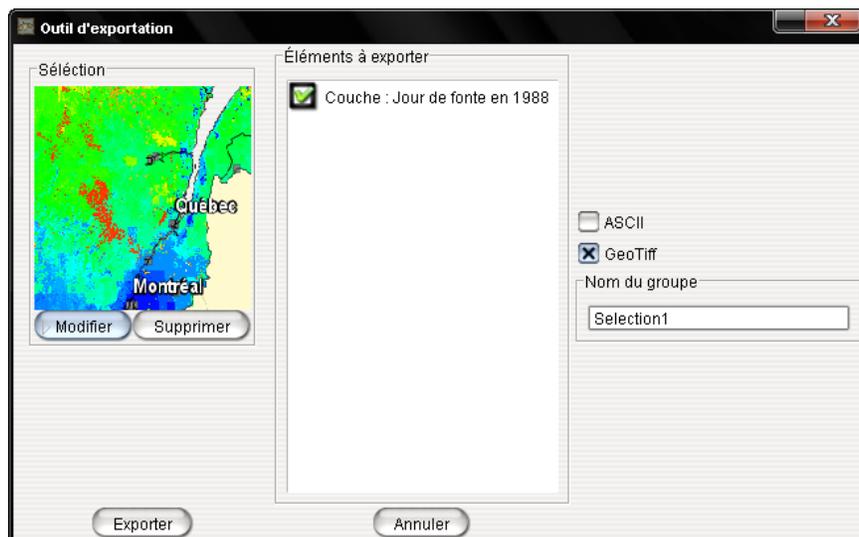


Figure 14 - Capture d'écran de la boîte de dialogue d'exportation

### *Le format GeoTiff*

Lorsque l'utilisateur n'a pas fait de sélection d'une région précise, l'ensemble de la couche active est copiée sur le disque dur tel quel. Dans le cas d'une sélection, le mécanisme est un peu plus complexe, au sens où il faut extraire les données qui correspondent à la sélection dans la couche. Pour ce faire il faut tout d'abord transformer les coordonnées géographiques de la sélection en coordonnées de pixel dans la couche en mémoire. Ensuite, corriger ces coordonnées si elles dépassent de l'image en mémoire, et enfin la recopier dans une nouvelle couche temporaire qui sera écrite sur le disque dur.

### *Le format Texte*

Selon le même principe que pour l'export en format GeoTiff, le format récupère la région à exporter. Ensuite l'Atlas fabrique un fichier texte qui contient la coordonnée géographique en X et en Y puis la valeur de la ou les variables sélectionnées.



## 4 DÉVELOPPEMENTS FUTURS

---

L'Atlas présenté dans ce rapport ne constitue qu'un prototype. Il s'agit d'un outil de démonstration de la pertinence du concept d'un Atlas interactif de la neige. Par conséquent, la base de données, les fonctionnalités et l'étendue du territoire devraient être bonifiés. Dans ce sens, plusieurs développements futurs sont proposés :

### 1) La base de données

Afin de saisir la climatologie de l'état du couvert nival, il est nécessaire d'augmenter la période d'observation. Celle-ci doit idéalement s'étaler sur une trentaine d'année. Toutefois, vue la limite sur la disponibilité des données satellitaires, cette période ne peut excéder les 25 ans. En effet, les données AVHRR et SSMI (son équivalent SMMR pour les années avant 1987) ne sont disponibles qu'à partir de 1984. Ceci représente une durée de 25 ce qui pourrait être considérée comme acceptable de point de vue climatologique. Par ailleurs, il serait intéressant d'inclure d'autres paramètres descriptifs du couvert nival tel que la date d'établissement du couvert en automne et la durée de la saison de couverture nivale. Pour ce faire, il faudrait adapter l'algorithme de cartographie de la neige de Chokmani *et al.* (2007) pour qu'il soit applicable à la période automnale. Au-delà la cartographie de la présence ou non de la neige, il serait également intéressant d'inclure des variables telles que la valeur maximale de l'équivalent en eau de la neige. Une série temporelle de cartes de cette variable est en développement par les équipes d'Hydro-Québec (Tapsoba *et al.*, 2005).

## 2) Les fonctionnalités

### a. Les variables dérivées et les paramètres statistiques

Dans la section des statistiques de base, il faudrait inclure d'autres valeurs centrales telles que le minimum, le maximum, et le mode. Par ailleurs, dans le cas de la prise en charge de d'autres paramètres descriptifs du couvert nival, il faudrait mettre à jour les outils statistiques pour gérer des variables multiples.

### b. L'outil de sélection et d'exportation

L'outil dans son état actuel de développement, le prototype ne permet pas de faire une sélection polygonale, ni une sélection par coordonnées géographiques. Une version future devrait le permettre. Ainsi, l'utilisateur pourrait exporter des données extraites au-dessous de certains points (stations de mesures) ou secteurs (bassins versants) du territoire.

## 3) Le portage sur Internet

La version actuelle de l'Atlas ne peut être déployé sur Internet. Ceci est principalement dû à sa conception qui ne permet pas sa diffusion sur l'Internet, et aussi au volume important de données manipulées dans l'Atlas. En effet, chaque image occupe environ 3,5 Mo, et par conséquent, le temps pour charger de telles images serait alors trop long. La solution consiste à faire calculer le rendu graphique par le serveur, et donc s'affranchir du téléchargement de toutes les couches vers le client, et ensuite de n'envoyer que le résultat du rendu. Cette solution permet donc d'économiser la bande passante, mais requiert des ressources

importantes du côté du serveur ainsi que l'accès à un serveur Internet cartographique.



## 5 RÉFÉRENCES

---

- Brown R. D., Braaten R. O. 1998. Spatial and temporal variability of Canadian monthly snow depths, 1946-1995. *Atmosphere-Ocean* 36: 37-45.
- Brown R. D., Brasnett B., Robinson D. 2003. Gridded North American monthly snow depth and snow water equivalent for GCM evaluation. *Atmosphere - Ocean* 41(1): 1-14.
- Brown R. D., Goodison B. E. 1996. Interannual variability in reconstructed Canadian snow cover, 1915-1992. *Journal of Climate* 9(6): 1299-1318.
- Chokmani K., Bernier M., Pâquet L. M., Goïta K., Roy A., Comtois-Boutet F., Turcotte M., Zhang Y., Forcier L.-C. 2007. Développement d'algorithmes pour le suivi par satellite de la couverture de neige au sol à l'échelle du bassin versant. Québec, Canada, Institut National de la Recherche Scientifique-Eau, Terre et Environnement: 73.
- CRYSYS 2007. State of the Canadian Cryosphere - Snow Atlas.  
<http://www.socc.ca/snow/atlas/index.cfm>.
- Derksen C., Walker A., LeDrew E., Goodison B. E. 2002. Utilizing satellite snow cover data for climatological analysis: A comparison of passive microwave and optically derived time series, 1978-1995 *International Geoscience and Remote Sensing Symposium (IGARSS)* 1: 217-219
- El Adlouni S., Ouarda T. B. M. J., Zhang X., Roy R., Bobée B. 2007. Generalized maximum likelihood estimators for the nonstationary generalized extreme value model *Water Resources Research* 43(3): W03410
- GeoTools 2008. The Open Source Java GIS Toolkit. <http://geotools.codehaus.org/>.

- Kitaev L., Førland E., Razuvaev V., Tveito O. E., Krueger O. 2005. Distribution of snow cover over Northern Eurasia. *Nordic Hydrology* 36(4-5): 311-319.
- Latifovic R., Trishchenko A. P., Chen J., Park W. B., Khlopenkov K. V., Fernandes R., Pouliot D., (...), Cihlar J. 2005. Generating historical AVHRR 1 km baseline satellite data records over Canada suitable for climate change studies. *Canadian Journal of Remote Sensing* 31(5): 324-346.
- OGC 2008. Open Geospatial Consortium, Inc. <http://www.opengeospatial.org/>.
- ÖNÖZ B., Bayazit M. 2003. The Power of Statistical Tests for Trend Detection. *Turkish Journal of Engineering & Environmental Sciences* 27: 247-251.
- Ressources Naturelles Canada 2007. L'atlas du Canada - Climat. 2007.  
<http://atlas.nrcan.gc.ca/site/francais/maps/environment/climate#snowcover>.
- Rikiishi K., Hashiya E., Imai M. 2004. Linear trends of the length of snow-cover season in the Northern Hemisphere as observed by the satellites in the period 1972-2000. *Annals of Glaciology* 38: 229-237.
- Tapsoba D., Fortin V., Anctil F., Haché M. 2005. Apport de la technique du krigeage avec dérive externe pour une cartographie raisonnée de l'Équivalent en Eau de la neige-application aux bassins de la rivière Gatineau. *Revue Canadienne de Génie Civil* 32(1): 289-297.
- Wikipédia 2008. Generalized extreme value distribution.  
[http://en.wikipedia.org/w/index.php?title=Generalized\\_extreme\\_value\\_distribution&oldid=223106910](http://en.wikipedia.org/w/index.php?title=Generalized_extreme_value_distribution&oldid=223106910).

# Annexes



## A.1 Le coin du développeur

L'Atlas adopte une conception particulière. Concrètement, cela se traduit par une classe centrale s'occupant de fournir les données et les moyens d'y accéder. Cette classe appelée *Data* agit aussi en qualité de passerelle entre les différents modules de l'Atlas qui ont besoin de communiquer entre eux.

Ce qui nous donne en quelque sorte une conception sous forme de noyau fonctionnel avec les différentes couches de l'interface se connectant dessus afin de tenir à jour l'affichage et manipuler les données.

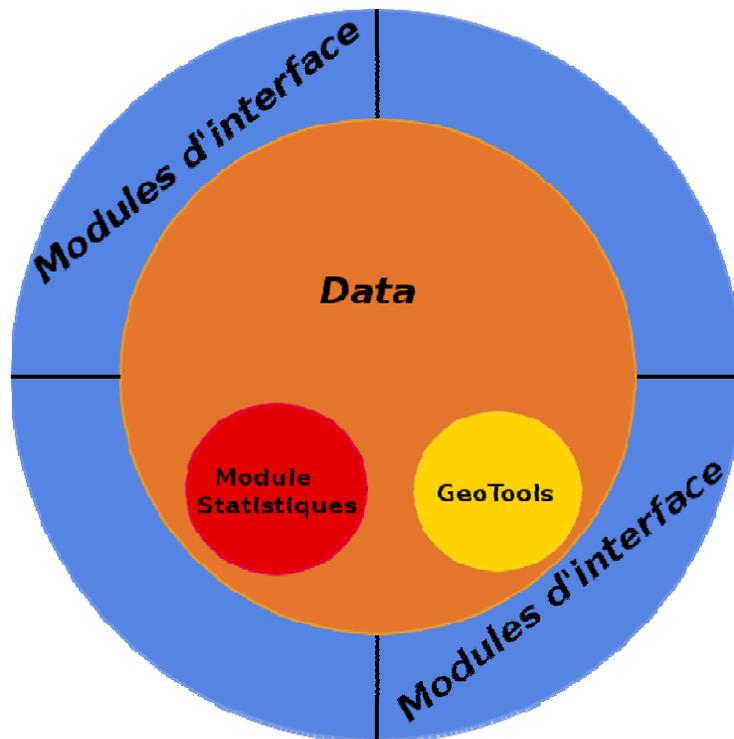


Figure 15 - Schéma de conception

Comme on le voit sur la Figure 15, la classe *Data* se repose à son tour sur le module de statistiques et la librairie GeoTools. Chaque module d'interface - comme la liste de couches - est réunie dans un package différent, et est liée à la classe *Data* pour avoir accès aux ressources.

De par le fait d'utiliser l'architecture SWING de Java pour développer et manipuler des composants graphiques, on a fait régulièrement appel au *Design Pattern Observer*, autrement appelé Modèle-Vue-Contrôleur (MVC). Cette méthode de conception sépare l'interface homme-machine (ihm) en un modèle (modèle de donnée), une vue (présentation, interface usager) et un contrôleur (logique de contrôle, gestion des événements, synchronisation).

## A.2 Exemples de code Java

Sont présentés ci-après quelques échantillons de codes particulièrement d'intérêt pour la poursuite du développement de l'Atlas.

### *A.2.1 La vue de la zone d'exportation*

Lorsque l'utilisateur veut exporter une zone sélectionnée, l'outil lui affiche cette zone. Pour cela il doit récupérer les coordonnées géographiques de la sélection et les transformer en coordonnées pixels du composant.

```

protected void paintComponent(Graphics g) {

    //Récupère le contexte graphique
    Graphics2D g2d = (Graphics2D)g;

    //Si on n'a pas le map on le dit
    if(this.data.getMap() == null){

        g2d.drawString("Carte non définie", 0, 10);

    }else{

        //Récupère el layer de sélection
        MapLayer lay = data.getSelection();

        if(lay != null){

            //Oui : On travaille juste sur la sélection
            BoundingBox bboxMap = null;

            BoundingBox bboxSelection = lay.getBounds();

            try{

                Envelope envMap = data.getMap().getRenderingStrategy().getMapArea();

                ReferencedEnvelope refEnv = new ReferencedEnvelope();

                refEnv.init(envMap);

                bboxMap = refEnv;

                //Si la sélection n'est pas visible entièrement
                if(!bboxMap.contains(bboxSelection)){

                    //On zoom en all

                    //Régler pour zoomer sur les geotiffs seulement
                    ReferencedEnvelope env = null;

                    //Cherche le layer de fond
                    for(int i = 0; i < data.getContext().getLayerCount();++i){

                        if((data.getContext().getLayers()[i].getTitle()).compareTo("Fond")==0){

                            env = data.getContext().getLayers()[i].getBounds();

                        }

                    }

                    data.getMap().getRenderingStrategy().setMapArea(env);
                }

            }

        }

    }
}

```

```

        while(data.getMap().getRenderingStrategy().isPainting()){
            this.repaint();
        }
    }
} catch(Exception e){
    e.printStackTrace();
}

//On définit le X, le Y, le width et le Height
double x1Sel = bboxSelection.getUpperCorner().getCoordinates()[0];
double y2Sel = bboxSelection.getUpperCorner().getCoordinates()[1];
double x2Sel = bboxSelection.getLowerCorner().getCoordinates()[0];
double y1Sel = bboxSelection.getLowerCorner().getCoordinates()[1];
Coordinate coordSelP2 = new Coordinate(x1Sel,y2Sel);
Point p2 = this.data.getMap().getRenderingStrategy().toComponentCoord(coordSelP2);
Coordinate coordSelP3 = new Coordinate(x2Sel,y2Sel);
Point p3 = this.data.getMap().getRenderingStrategy().toComponentCoord(coordSelP3);
Coordinate coordSelP4 = new Coordinate(x2Sel,y1Sel);
Point p4 = this.data.getMap().getRenderingStrategy().toComponentCoord(coordSelP4);

int x = p3.x;
int y = p3.y;
int width = p2.x - p3.x;
int height = p4.y - p3.y;

try{

    BufferedImage img = this.data.getMap().getRenderingStrategy().getSnapshot();
    //Vérifie la validité des coordonnées
    if(x < 0)
        x = 0;
}

```

```

        if(y < 0)
            y = 0;
        if(x+width > img.getWidth())
            width = img.getWidth()-x;
        if(y + height > img.getHeight())
            height = img.getHeight() - y;
        //Et on affiche
        BufferedImage resImg = scale(img.getSubimage(x, y, width,
height),this.getWidth(),this.getHeight());
        g.drawImage(resImg, 0, 0, resImg.getWidth(), resImg.getHeight(), null);
    } catch(Exception e){
        e.printStackTrace();
    }
} else{
    //Non : on travaille sur la totalité de la région
    BufferedImage img =
scale(this.data.getMap().getRenderingStrategy().getSnapshot(),this.getWidth(),this.getHeight());
    g.drawImage(img, 0, 0, img.getWidth(), img.getHeight(), null);
}
}
}
}

```

### A.2.2. Le chargement d'une couche

Exemple du code nécessaire au chargement d'une couche de données concernant le couvert nival.

Le mécanisme général est le suivant :

- Le système demande le chargement d'une date précise. Si cette date a déjà été chargée, le système se replace dans le contexte de cette date et attend qu'on vienne récupérer les informations
- Si la carte n'a pas déjà été chargée, le système la charge et se positionne dans le contexte de la date qui viens d'être chargée, et attend qu'on vienne récupérer les informations

#### A.2.2.1. Chargement d'un contexte

```
/**
 * @param année Année source
 * @param mois Mois de l'année, entre 1 et 12
 * @param jour Jour du mois : entre 1 et 31 (selon le mois et l'année)
 * @return booléen VRAI si le fichier est chargé, FAUX sinon
 * @throws IOException
 * Lire un fichier GéoTIFF
 *
 *     -> Commence par vérifier si les paramètres sont bons. Si jamais l'année fournie et/ou le jour fournit n'est
 pas correct
 *
 *     alors la fonction s'arrête et retourne FAUX.
 *
 *     -> L'exception concerne les erreurs de lecture système
 */
public boolean setContextGeoTiff(int annee,int mois, int jour) throws IOException{
    String year;
    String month;
    String day;
    String path;
    //Si on était dans un contexte d'usage temporaire
    if(this.tmpUse){
        this.clean();
    }
}
```

```

}

this.tmpUse = false;

//Vérifie que le contexte n'existe pas déjà !!!
String key = new Integer(annee).toString();
key = key + "-" + new Integer(mois).toString();
key = key + "-" + new Integer(jour).toString();
if(this.listGeoTiffs.get(key) == null){

    //Vérifie les paramètres de mois et de jours

    if(mois > 12 || mois < 1){

        System.out.println("setContextGeoTiff : Paramètre Mois hors limites");

        return false;

    }

    //Année bissextile ? (Pour le mois de février)

    boolean bBisextile = false;

    if(annee % 4 == 0){

        bBisextile = true;

    }

    //Switch selon le mois

    boolean errDay = false;

    switch(mois){

        case 1:

        case 3:

        case 5:

        case 7:

        case 8:

        case 10:

        case 12:

            if(jour > 31 || jour < 1)

                errDay = true;

            break;

        case 4:

```

```

        case 6:

        case 9:

        case 11:
            if(jour > 30 || jour < 1)
                errDay = true;
            break;
        case 2:
            if(bBisextile){
                if(jour > 29 || jour < 1)
                    errDay = true;
            }else{
                if(jour > 28 || jour < 1)
                    errDay = true;
            }
    }

    //Erreur sur le jour ?
    if(errDay){
        System.out.println("setContextGeoTiff : Paramètre Jour hors limites");
        return false;
    }

    //Vérifie que l'on a bien l'année en stock
    File dir = new File("./Res/GeoTIFF/CouvertNival/");
    String[] childrens = dir.list();

    //Parcours la liste de répertoires afin de trouver celui de l'année indiquée
    boolean bTemp = false;
    year = (new Integer(annee)).toString();
    for(int i = 0; i < childrens.length; ++i){
        if(childrens[i].compareTo(year) == 0){
            //On a trouvé le répertoire
            bTemp = true;
        }
    }

```

```

    }

    //Si on n'a pas trouvé le répertoire
    if (!bTemp){
        System.out.println("setContextGeoTiff : Paramètre Année hors limites -> L'année demandée
n'est pas disponible");

        return bTemp;
    }

    //Rentre dans ce répertoire et trouve le jour
    dir = new File("./Res/GeoTIFF/CouvertNival/" + year + "/");
    childrens = dir.list();

    //Construit le nom de fichier
    if(mois < 10)
        month = "0" + (new Integer(mois)).toString();
    else
        month = (new Integer(mois)).toString();

    if(jour < 10)
        day = "0" + (new Integer(jour)).toString();
    else
        day = (new Integer(jour)).toString();

    //Gère l'an 2000 !!!
    if(annee < 2000){
        year = (new Integer(annee-1900)).toString();
    }

    //Nom du fichier
    path = year + month+day+"_fusion.tif";

    //Parcours la liste des fichiers du répertoire pour trouver le fichier demandé en entrée
    bTemp = false;
    for(int i = 0; i < childrens.length;++i){
        //Construit
        if(childrens[i].compareTo(path) == 0){

```

```

        //On a trouvé le répertoire
        bTemp = true;
    }
}
//Gestion de l'erreur : a-t-on trouvé le fichier ?
if(!bTemp){
    System.out.println("setContextGeoTiff : Le jour et le mois demandé n'existe pas pour cette
année !");
    return bTemp;
}
//Ouvre le fichier
path = dir.getPath()+"/" + path;
File tif = new File(path);
//Extrait le GeoTIFF
GeoTiffReader gtReader;
try{
    gtReader = new GeoTiffReader(tif);
} catch (DataSourceException ex) {
    ex.printStackTrace();
    return false;
}
//Coordonnées
this.crsGeoTIFF = gtReader.getCrs();

try{
    this.gcGeoTIFF = (GridCoverage2D)gtReader.read(null);
} catch (IOException ex) {
    ex.printStackTrace();
    return false;
}
//Copie le geotiffreader

```

```

this.geoRead = gtReader;

//Copie la clé
this.name = key.substring(2);

//Sauvegarde le contexte
this.listGeoTiffs.put(key, gtReader);

} else {

//Le contexte est déjà présent, on va se contenter de le charger
GeoTiffReader gtReader = this.listGeoTiffs.get(key);
this.crsGeoTIFF = gtReader.getCrs();

try {

        this.gcGeoTIFF = (GridCoverage2D)gtReader.read(null);

} catch (IOException ex) {

        ex.printStackTrace();

        return false;

}

//Copie le geotiffreader
this.geoRead = gtReader;

//Copie la clé
this.name = key.substring(2);

}

//Travail correctement effectué

return true;

}

```

### A.2.2.2. Création de la couche à partir du contexte

```
/**
 * Créé un layer d'après un geoTiff NOAA-SSMI fusion
 * @return layer
 */
public MapLayer getLayerFromCurrentContext(){
    MapLayer lay = null;
    if(this.geoRead == null || this.gcGeoTIFF == null || this.crsGeoTIFF == null){
        System.out.println("Impossible de créer la couche : pas de contexte définit");
        return null;
    }

    //Construit le colormap
    ColorMap colorMap = new ColorMapImpl();
    colorMap.setType(ColorMap.TYPE_VALUES);
    colorMap.addColorMapEntry(ColorMapFactory.createColorMapEntry("Pas de données",0,new Color(245,
250,245),0.0));
    colorMap.addColorMapEntry(ColorMapFactory.createColorMapEntry("Neige",255,new Color(230, 230, 255),1.0));
    colorMap.addColorMapEntry(ColorMapFactory.createColorMapEntry("Non neige",50,new Color(0,128,0),1.0));

    //Construit le style
    StyleBuilder sb = new StyleBuilder();
    RasterSymbolizer symb = sb.createRasterSymbolizer(colorMap,1.0);
    Style rasterStyle = sb.createStyle( symb );

    //Créé un layer
    try{
        lay = new DefaultMapLayer(this.gcGeoTIFF,rasterStyle);
        lay.setTitle("Couv. nival " + this.name);
        lay.setVisible(true);
        lay.setQuery(Query.ALL);
    }
```

```

    } catch(Exception ex){
        System.out.println(ex.toString());
    }
    return lay;
}

```

### A.2.2.3. Encapsulation d'une couche GeoTools

Ce processus est appelé à chaque ajout d'une couche dans le contexte de la carte principale.

```

public void layerAdded(MapLayerListEvent e){
    //On vérifie s'il n'a pas été déjà ajouté auparavant (Création préalable possible lors de l'appel de getExtendedLayer())
    for(ExtendedLayer tmp:this.lstLayers){
        if(tmp.getLayer().equals(e.getLayer())){
            //On l'a déjà créé au préalable, on stoppe le processus de création
            return;
        }
    }
    //Ajoute le layer à la liste des extended layers
    ExtendedLayer rastLay = new ExtendedLayer(e.getLayer());
    //Définit le colormap
    if(e.getLayer().getStyle().getFeatureTypeStyles()[0].getRules()[0].getSymbolizers()[0] instanceof
RasterSymbolizerImpl){
        rastLay.setCMap(((RasterSymbolizerImpl)e.getLayer().getStyle().getFeatureTypeStyles()[0].getRules()[0].getSymbolizers()[0]).getColorMap());
    }
    //Crée la légende
    LegendList leg = new LegendList(this);
}

```

```
rastLay.setLegende(leg);  
  
this.lstLayers.add(rastLay);  
  
}
```

### A.3 L'Outil Converter

Cet outil a été développé pour pouvoir exploiter les données en sortie de l'outil Matlab® directement dans l'Atlas. Son rôle consiste à récupérer les données en format texte de Matlab®, et de produire des images géo-référencées directement affichables par l'Atlas.

L'utilisateur doit donc - pour mettre à jour ses données de moyenne, écart-type, et coefficient de variation - lancer les fonctions Matlab®, et ensuite lancer l'outil *Converter*.

Voici une capture d'écran de la sortie de l'outil *Converter* :

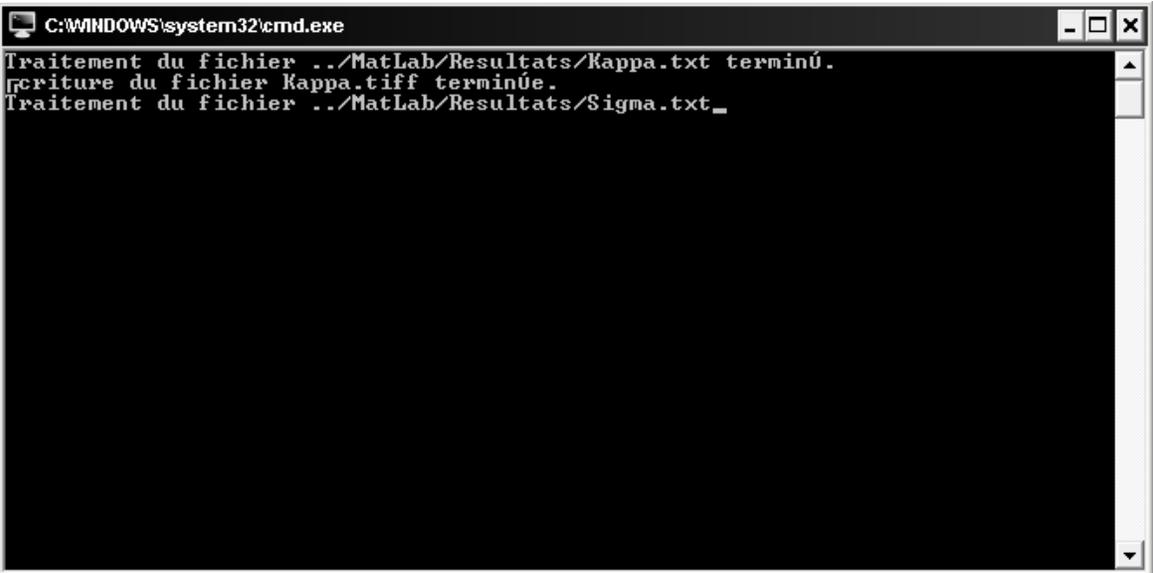


Figure 16 - Capture écran de la sortie de l'outil *Converter*

#### A.4 DVD du prototype de l'Atlas

La consultation de l'Atlas nécessite que la plateforme Java - Version 6 mise à jour 7 (Build 1.6.0\_07-b06) ou une version plus récente, soit installée sur la machine.

