

LogiLab : AccessFileGenerator
– Guide utilisateur

Alexandre Morneau

Consortium de recherche FORAC

Table des matières

| | |
|---|----|
| 1 Description du AccessFileGenerator de LogiLab | 3 |
| 2 Utilisation du AccessFileGenerator de LogiLab | 3 |
| 2.0 Prérequis de l'application | 3 |
| 2.1 Structure du fichier Excel | 5 |
| 2.1.1 BUTYPE_info | 5 |
| 2.1.2 ProblemProperties | 6 |
| 2.1.3 Periods | 6 |
| 2.1.4 Products | 6 |
| 2.1.5 BusinessUnits | 7 |
| 2.1.6 Processes | 8 |
| 2.1.7 ProcessIN | 8 |
| 2.1.8 ProcessOut | 9 |
| 2.1.9 InventoryCapacity | 9 |
| 2.1.10 Supply | 10 |
| 2.1.11 Demand | 11 |
| 2.1.12 Flows | 11 |
| 2.2 Procédure de génération du fichier d'entrée Access | 14 |
| 2.3 Résolution de problèmes | 16 |
| 2.3.1 Could not write into Access file, please install 'Microsoft Access Database Engine 2010 Redistributable' for X64 architecture | 16 |

1 Description du AccessFileGenerator de LogiLab

Bien que le fichier d'entrée Access, avec ses formulaires, soit la meilleure façon de parcourir, visualiser et modifier un cas LogiLab, il n'est pas particulièrement efficace pour importer de grandes quantités de données.

C'est spécifiquement pour faciliter et accélérer l'importation de grandes quantités de données que le AccessFileGenerator a été développé. Il suffit de copier-coller les données à importer dans un fichier Excel de saisi contenant une structure spécifique et l'application l'importera automatiquement dans un fichier d'entrée Access fonctionnel, prêt à être importé dans LogiLab.

2 Utilisation du AccessFileGenerator de LogiLab

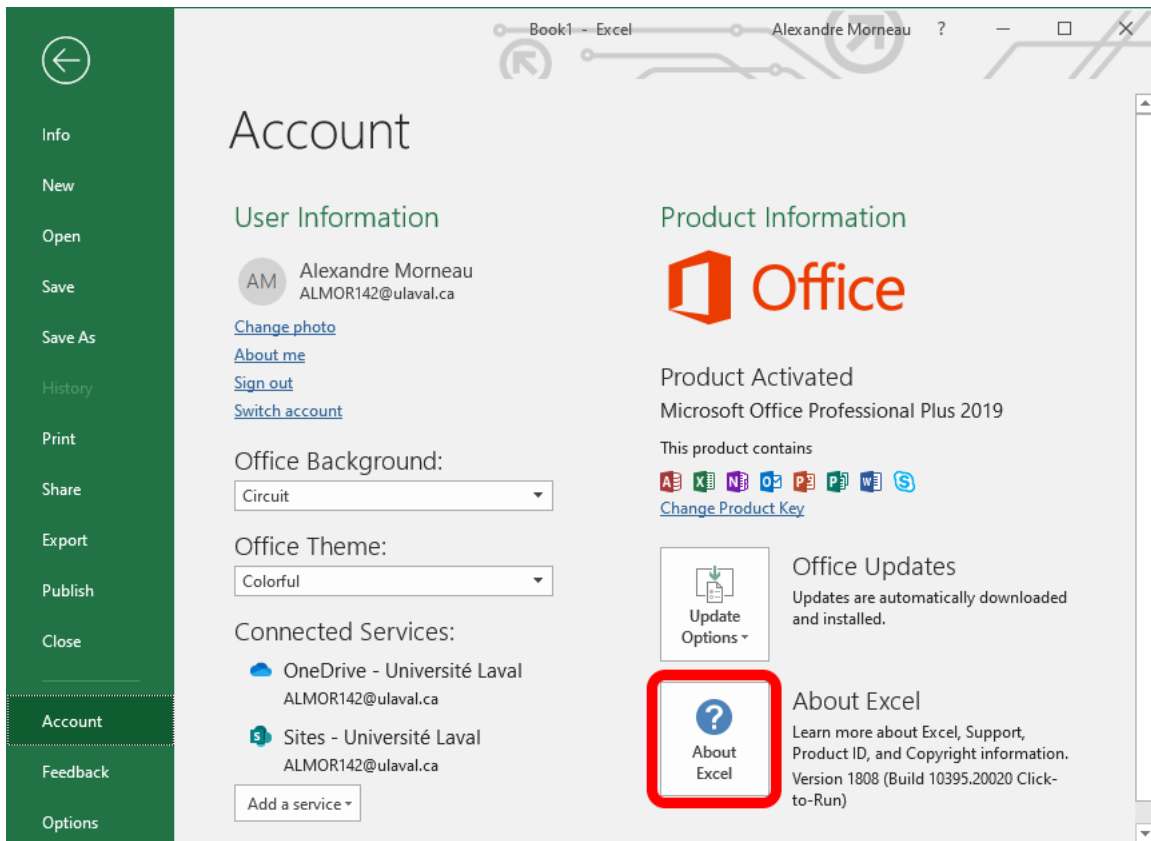
Cette section du document présente la structure des différentes sections du fichier Excel utilisé pour préparer les données à l'importation. Ensuite, la procédure pour générer un fichier d'entrée Access à partir du fichier Excel sera présentée.

2.0 Prérequis de l'application

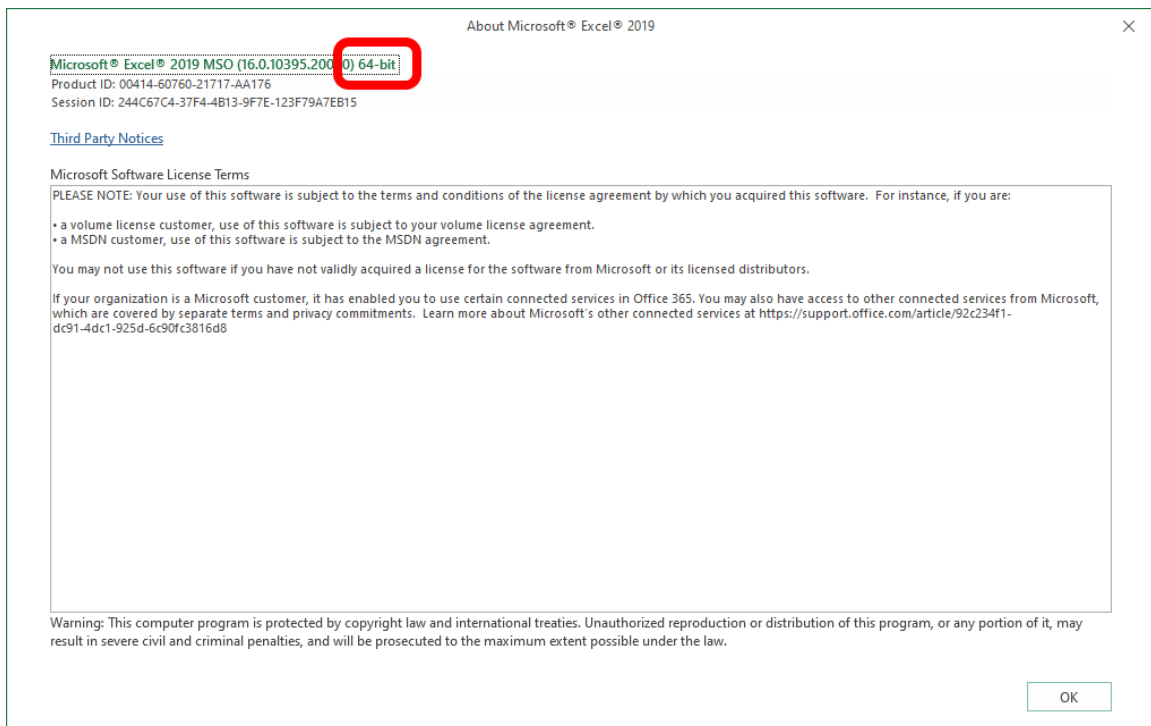
Pour fonctionner, cette application doit interagir avec une installation fonctionnelle et compatible de Microsoft Excel et de Microsoft Access.

2.0.1 Microsoft Excel

Pour permettre à l'application de lire le fichier d'entrée de format Microsoft Excel, une installation de Microsoft Excel compatible doit être présente sur votre machine. Pour être compatible, Microsoft Excel doit avoir le même « bitage » (32 bit ou 64 bit) que votre AccessFileGenerator. Pour vérifier le « bitage » de votre Ms. Excel, dans le menu « File / Fichier », sélectionnez l'onglet « Account / Compte », puis appuyez sur le bouton « About Excel / À propos d'Excel ».



Cela ouvre la fenêtre d'information de l'application, de laquelle vous allez pouvoir tirer le nombre de bits utilisé.



Dans cet exemple, on voit que Microsoft Excel est installé en 64 bits, donc il faudra utiliser le AccessFileGenerator 64 bits pour assurer la compatibilité.

2.0.2 Microsoft Access

Pour permettre d'écrire rapidement dans un fichier Access, l'application utilise une librairie appelée le « Data Access Object ». Ainsi, pour fonctionner sur votre poste vous devez avoir installé Microsoft Access avec le « bitage » que celui identifié à l'étape 2.0.1. Alternativement, vous pouvez installer ce qu'on appelle un « Access Redistribuable ». Notez que si vous avez déjà une version de Microsoft Access d'installé, vous devrez installer un redistribuable d'une année différente. Ainsi, vous pouvez télécharger un redistribuable Access, x32 ou x64, sur un des liens suivants :

- Version 2016 :
<https://www.microsoft.com/en-us/download/details.aspx?id=54920>
- Version 2010 :
<https://www.microsoft.com/fr-ca/download/details.aspx?id=13255>

2.1 Structure du fichier Excel

L'application vient toujours avec un exemple de fichier d'importation nommé « ImportAccess.xlsx ». Le fichier contient les feuilles suivantes :

- BUTYPE_info
- ProblemProperties
- Periods
- Products
- BusinessUnits
- Processes
- ProcessIN
- ProcessOUT
- InventoryCapacity
- Supply
- Demand
- Flows

2.1.1 BUTYPE_info

Cette feuille contient les types de BusinessUnits supportés par LogiLab. Elle sert de référence pour aider l'utilisateur à sélectionner les types de ces BusinessUnits parmi les choix disponibles. Cette feuille ne doit pas être modifiée.

2.1.2 ProblemProperties

Cette feuille contient 2 colonnes :

- **PropertyName** : Nom des propriétés du problème à définir. (ne pas modifier)
- **PropertyValue** : Valeur des propriétés du problème à définir.

Cette feuille permet de saisir les propriétés du problème suivantes :

- **ProblemName** : le nom du problème
- **Description** : (facultative)
- **SecondaryValueLabel** : (facultatif) À utiliser si l'on désire ajouter un deuxième élément à mettre en relation avec l'aspect financier. On y inscrit le nom de l'autre élément comme par exemple, « Carbone ». Cette valeur doit être définie pour que les champs « SecondaryValue » des autres feuilles soient considérés.

2.1.3 Periods

Cette feuille permet de saisir l'aspect temporel du problème. On y saisit une ou plusieurs périodes. Chacune de ces périodes, définies par une date de début et une date de fin, doivent être de durées égales. Chacune de ces périodes peut être ajoutée à un ou à plusieurs groupes de périodes. La feuille contient un minimum de 4 colonnes :

- **Code** : Le numéro de la période. Il doit être un nombre entier plus grand que 0, et augmenter de 1 pour chaque période suivante.
- **StartDate** : La date de début de chaque période. Doit respecter le format « date » ou « datetime » d'Excel.
- **EndDate** : La date de fin de chaque période. Doit respecter le format « date » ou « datetime » d'Excel et évidemment être plus grand que le StartDate.
- **PeriodSetName->** : Cette colonne sert à indiquer que les colonnes suivantes (à sa droite) définissent des groupes de période. Les groupes de périodes peuvent être utilisés dans les autres feuilles pour éviter d'avoir à copier-coller des données qui se répètent à plusieurs périodes. Pour créer un groupe de période, il suffit d'ajouter le nom du groupe comme titre de colonne sur la ligne 1, à droite de la colonne « PeriodSetName-> ». Ensuite, pour ajouter des périodes dans le groupe, il suffit d'ajouter un « x » dans la colonne du groupe sur chaque ligne des périodes que vous désirez ajouter.

2.1.4 Products

Cette feuille permet de définir l'ensemble des produits qui seront utilisés dans le réseau logistique modélisé. La feuille contient 3 colonnes :

- **ProductCode** : Le nom du produit. Il doit être unique et il sera utilisé dans les autres feuilles.

- ProductType : Le type de produit, utilisé pour regrouper et catégoriser les produits. Assurez-vous de toujours utiliser exactement la même syntaxe pour un même type.
- UOM : L'unité de mesure utilisée pour quantifier le produit. Assurez-vous de toujours utiliser exactement la même syntaxe pour une même unité. Il sera utilisé dans les autres feuilles

2.1.5 BusinessUnits

Cette feuille permet de définir l'ensemble des BusinessUnits (BU) qui constituent les nœuds de votre réseau logistique. Cette feuille sert également à définir les capacités de productions des usines du réseau. La feuille contient les colonnes suivantes :

- BusinessUnitName : Le nom du BU. Il doit être unique et il sera utilisé dans les autres feuilles.
- BusinessUnitType : Le type du BU. Il doit obligatoirement respecter la syntaxe exacte des types énumérés à la colonne B de la feuille « BUTYPE_info ». Le type du BU déterminera son rôle dans le réseau. Par exemple, seuls les fournisseurs peuvent fournir du supply, seules les productions peuvent avoir une production avec des process et seuls les clients peuvent avoir une demand. Le type déterminera aussi l'icône qui représentera le BU dans l'interface Web de LogiLab.
- InventoryCost : Le coût d'inventaire du BU. Pour l'explication détaillée de cette valeur, voir la section 3.3.3 du guide d'utilisateur LogiLab.
- Latitude et Longitude : La position géographique du BU. Utilisées pour afficher le BU sur la carte géographique. Elle est aussi utilisée pour calculer la distance des flux (flows) à vol d'oiseau automatiquement si celle-ci n'est pas définie.
- X et Y : La position x y du BU. Utilisées pour afficher le BU dans un schéma conceptuel. (facultatif)
- ProductionCapacity_Type : L'unité utilisée pour exprimer la capacité de production du BU. Cette valeur est uniquement utilisée pour les BU de type Production, elle doit rester vide dans les autres cas. Actuellement, la seule valeur supportée est « HOURS ».
- ProductionCapacity_Min : Le nombre d'heure minimale de production que le BU doit absolument effectuer durant une période. Cette valeur est uniquement utilisée pour les BU de type Production, elle doit rester vide dans les autres cas.
- ProductionCapacity_Max : Le nombre d'heure maximale de production que le BU peut effectuer durant une période. Cette valeur est uniquement utilisée pour les BU de type Production, elle doit rester vide dans les autres cas.
- GroupTags : La liste des GroupTags (étiquettes de regroupement), séparés par des virgules, s'appliquant à ce BU. Les GroupTags permettent d'identifier un BU comme faisant parti d'un group. Les BU regroupés peuvent être ainsi utilisés dans un script Python pour effectuer certains calculs spécifiques ou encore pour

ajouter des contraintes. Pour plus de détails, voir la section 3.1.4 du guide d'utilisateur LogiLab.

2.1.6 Processes

Cette feuille permet de définir l'ensemble des processus de production des BU de type production. Dans LogiLab, un processus est en quelque sorte une recette de fabrication. Pour chaque processus, on définit les intrants dans la feuille ProcessIN et les extrants dans la feuille ProcessOut. Cette feuille contient les 5 colonnes suivantes :

- BusinessUnitName : Le nom du BU auquel appartient ce processus. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- ProcessName : Le nom du processus de production. Il doit être unique et il sera utilisé dans les feuilles ProcessIN et ProcessOut.
- CapacityUsage : Le nombre d'heure de production nécessaire pour effectuer le processus une fois. Ce champ fait référence aux champs ProductionCapacity de la feuille BusinessUnits. En effet, la somme du CapacityUsage de tous les processus effectués pour un BU donné pour une période donnée devra toujours se retrouver entre les valeurs du ProductionCapacity_Min et du ProductionCapacity_Max du BU.
- Cost : Le coût de production à comptabiliser à chaque fois que l'on exécute le processus une fois.
- SecondaryValue : Une valeur secondaire à comptabiliser à chaque fois que l'on exécute le processus une fois. Le type et l'unité de cette valeur peut être n'importe quoi, tant qu'elle est la même que tous les autres SecondaryValue définies dans le même fichier. Habituellement utilisé pour le carbone.
- Discrete? : Permet de définir si le processus est discret ou si il est continu. Par défaut, les processus de LogiLab sont tous continus. C'est-à-dire qu'ils peuvent être exécutés partiellement. Par exemple, pour la moitié du CapacityUsage et du Cost, il va utiliser la moitié des ressources entrantes pour produire la moitié des ressources sortantes. Dans ce cas, on parle d'un problème d'optimisation linéaire. LogiLab peut aussi être utilisé pour résoudre le problème en mode entier mixtes « Mixed Integer Programming » ou « MIP ». En mode MIP, LogiLab peut utiliser des variables entières. Donc, si LogiLab est en mode MIP, les processus identifiés comme étant discrets devront être exécuté un nombre entier de fois. Donc, il ne sera plus permis d'exécuter partiellement ces processus.

2.1.7 ProcessIN

Cette feuille permet de définir l'ensemble des ressources entrantes pour chacun des processus définis dans la feuille Processes. Cette feuille contient les 4 colonnes suivantes :

- BusinessUnitName : Le nom du BU auquel appartient le processus. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.

- **ProcessName** : Le nom du processus de production pour lequel on veut définir une entrée. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Processes.
- **ConsumedProduct** : Le nom du produit consommé par ce processus. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products.
- **ConsumedQty** : La quantité de produit qui sera consommé à chaque exécution du processus. L'unité utilisée est celle définie pour le produit dans la feuille Products.

2.1.8 ProcessOut

Cette feuille permet de définir l'ensemble des ressources sortantes pour chacun des processus définis dans la feuille Processes. Cette feuille contient les 4 colonnes suivantes :

- **BusinessUnitName** : Le nom du BU auquel appartient le processus. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- **ProcessName** : Le nom du processus de production pour lequel on veut définir une sortie. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Processes.
- **ProducedProduct** : Le nom du produit construit par ce processus. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products.
- **ProducedQty** : La quantité de produit qui sera produite à chaque exécution du processus. L'unité utilisée est celle définie pour le produit dans la feuille Products.

2.1.9 InventoryCapacity

Cette feuille permet de définir les capacités de stockage de chaque BU. Par défaut, un BU ne peut rien stocker. Soit on définit une capacité globale pour l'ensemble des produits, soit on définit une capacité spécifique pour chacun des produits que l'on désire stocker individuellement. Il n'est actuellement pas recommandé d'utiliser une combinaison de capacité globale et de capacités par produits. Cette feuille contient les 4 colonnes suivantes :

- **BusinessUnitName** : Le nom du BU pour lequel on veut définir une capacité de stockage. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- **ProductCode** : Le nom du produit pour lequel on désire définir une capacité de stockage. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products. Pour définir une capacité de stockage globale, laissez ce champ vide.
- **Min** : La quantité minimum de produit qui doit être conservée en inventaire en tout temps. L'unité utilisée est celle associée au produit dans la feuille Products.
- **Max** : La quantité maximum de produit qui peut être conservée en inventaire en tout temps. L'unité utilisée est celle associée au produit dans la feuille Products. Pour une capacité illimitée, utilisez la valeur NULL.

2.1.10 Supply

Cette feuille permet de définir les différentes quantités de produits pouvant être introduites dans le réseau logistique par les différents BU aux différentes périodes du problème. Cette feuille contient les 7 colonnes suivantes :

- BusinessUnitName : Le nom du BU pour lequel on veut définir une capacité de fournir un produit. Le BU en question doit obligatoirement être de type supplier. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- PeriodSetName : Le nom du groupe de période pour lequel cette ligne s'applique. Cela signifie que cette capacité de fournir un produit s'appliquera pour toutes les périodes contenues dans le groupe de période. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Periods. Alternativement, si vous n'avez pas plusieurs périodes à gérer, vous pouvez aussi utiliser directement le Code de la période tel que définie dans la feuille Periods.
- IsMandatory : Uniquement deux valeurs sont permises pour ce champ :
 - FALSE : si le fournisseur n'est pas obligé d'introduire, au total ou en partie, ces produits dans le réseau logistique, il le fera seulement si c'est optimal.
 - TRUE : si le fournisseur est obligé d'introduire la totalité de ces produits dans le réseau logistique, même si cela n'est pas optimal. Notez que IsMandatory et IsInfinite ne peuvent pas être TRUE pour la même ligne.
- IsInfinite : Uniquement deux valeurs sont permises pour ce champ :
 - FALSE : Si le fournisseur ne peut pas fournir plus de produit que la quantité définie dans la colonne QuantityPerPeriod.
 - TRUE : Si le fournisseur ignore la colonne QuantityPerPeriod et n'a pas de limite maximale à la quantité de produit qu'il peut introduire dans le réseau logistique. Notez que IsMandatory et IsInfinite ne peuvent pas être TRUE pour la même ligne.
- Product : Le nom du produit que le BU pourra fournir. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products.
- Price : Le coût unitaire facturé à chaque fois qu'un produit est fourni.
- SecondaryValue : Une valeur secondaire à comptabiliser à chaque fois que l'on fournit un produit. Le type et l'unité de cette valeur peut être n'importe quoi, tant qu'elle est la même que tous les autres SecondaryValue définies dans le même fichier. Habituellement utilisé pour le carbone.
- QuantityPerPeriod : La quantité maximale du produit pouvant être fournie à chaque période ciblée.

2.1.11 Demand

Cette feuille permet de définir les différentes quantités de produits pouvant être retirées du réseau logistique par les différents BU aux différentes périodes du problème. Cette feuille contient les 7 colonnes suivantes :

- BusinessUnitName : Le nom du BU pour lequel on veut définir une demande de produit. Le BU en question doit obligatoirement être de type client. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- PeriodSetName : Le nom du groupe de période pour lequel cette ligne s'applique. Cela signifie que cette capacité d'acheter un produit s'appliquera pour toutes les périodes contenues dans le groupe de période. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Periods. Alternativement, si vous n'avez pas plusieurs périodes à gérer, vous pouvez aussi utiliser directement le Code de la période tel que définie dans la feuille Periods.
- IsMandatory : Uniquement deux valeurs sont permises pour ce champ :
 - FALSE : si il n'est pas obligatoire de combler entièrement la demande du client si ce n'est pas optimal de le faire.
 - TRUE : si la demande du client doit obligatoirement être comblée, même si cela n'est pas optimal. Notez que IsMandatory et IsInfinite ne peuvent pas être TRUE pour la même ligne.
- IsInfinite : Uniquement deux valeurs sont permises pour ce champ :
 - FALSE : Si le client ne peut pas acheter plus de produit que la quantité définie dans la colonne QuantityPerPeriod.
 - TRUE : Si le client ignore la colonne QuantityPerPeriod et n'a pas de limite maximale à la quantité de produit qu'il peut retirer du le réseau logistique. Notez que IsMandatory et IsInfinite ne peuvent pas être TRUE pour la même ligne.
- Product : Le nom du produit que le BU pourra fournir. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products.
- Price : Le revenu unitaire reçu à chaque fois qu'un produit est acheté par le client.
- SecondaryValue : Une valeur secondaire à comptabiliser à chaque fois qu'un client achète une unité d'un produit. Le type et l'unité de cette valeur peut être n'importe quoi, tant qu'elle est la même que tous les autres SecondaryValue définies dans le même fichier. Habituellement utilisé pour le carbone.
- QuantityPerPeriod : La quantité maximale du produit pouvant être acheté par le client pour chacune des périodes ciblées.

2.1.12 Flows

Cette feuille permet de définir les différents flux permettant de faire circuler les différents produits entre les différents BU. La feuille contient un minimum de 12 colonnes :

- Origin : Le nom du BU d'origine du flux. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- Destination : Le nom du BU de destination du flux. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille BusinessUnits.
- TransportationMode : Le type de moyen de transport, utilisé pour regrouper et catégoriser les flux. Assurez-vous de toujours utiliser exactement la même syntaxe pour un même moyen de transport.
- DelayDays : Le nombre de jour de transport entre l'origine et la destination. Si ce délai est plus grand que la durée d'une période multipliée par le paramètre d'optimisation « *FlowDiscretizationDelayPercentage* », la quantité de produit déplacée par le flux arrivera à la période suivante.
- Cost/UOM/Km : Le coût de transport par unité de produit transportée par KM. Par exemple, si on veut déplacer 50 unités sur 100 KM avec un Cost/UOM/KM de 0.1\$/unité/KM, un montant de 500\$ sera facturé. Si on veut plutôt définir un coût par unité uniquement, il suffit de saisir une valeur de 1 dans la colonne DistanceKM.
- SecondaryValue : Une valeur secondaire à comptabiliser à chaque fois qu'une unité d'un produit est déplacée d'un KM. Le type et l'unité de cette valeur peut être n'importe quoi, tant qu'elle est la même que tous les autres SecondaryValue définies dans le même fichier. Habituellement utilisé pour le carbone.
- DistanceKM : Le nombre de KM du flux. Cette valeur est utilisée pour calculer le coût de transport des unités de produit voyageant sur ce flux. Si vous laissez ce champ vide, LogiLab calculera automatiquement la distance à vol d'oiseau.
- UnitOfMeasure : L'unité des produits pouvant être transportés par ce flux. Pour transporter des produits ayant une autre unité de mesure, un nouveau flux devra être créé. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Products.
- LoadSize : Permet de définir si le flux est discret ou s'il est continu. Par défaut, les flux de LogiLab sont tous continus. C'est-à-dire qu'ils ne gèrent pas la notion de chargements pleins ou partiels. Les frais de transports sont calculés par unités transportées, au lieu de par chargement. Dans ce cas, on parle d'un problème d'optimisation linéaire. LogiLab peut aussi être utilisé pour résoudre le problème en mode entier mixtes « Mixed Integer Programming » ou « MIP ». En mode MIP, LogiLab peut utiliser des variables entières. Donc, si LogiLab est en mode MIP, on peut définir une valeur de LoadSize et LogiLab sera obligé de transporter des chargements aillant exactement le même nombre d'unité que dans le LoadSize. Si vous prévoyez utiliser cette valeur, assurez-vous d'avoir des flux avec des chargements de petites tailles, sinon, certains produits n'ayant pas la quantité nécessaires ne pourront jamais être transportés.
- CapacityPeriodSet : Le nom du groupe de période pour lequel ce flux s'applique. Cela signifie que ce flux (incluant sa capacité) s'appliquera pour toutes les

périodes contenues dans le groupe de période. Assurez-vous d'utiliser exactement la même syntaxe que dans la feuille Periods. Alternativement, si vous n'avez pas plusieurs périodes à gérer, vous pouvez aussi utiliser directement le Code de la période tel que définie dans la feuille Periods.

- CapacityMin : La quantité minimum d'unité de produit qui doit obligatoirement transiger par ce flux pour chaque période données. Ce champ peut rester vide.
- CapacityMax : La quantité maximum d'unité de produit qui peut transiger par ce flux pour chaque période données. Pour une capacité illimitée, ce champ peut rester vide.
- CapacityProducts-> : Cette colonne sert à indiquer que les colonnes suivantes (à sa droite) définissent quels produits sont permis dans le flux. Ainsi, les colonnes suivantes contiennent des noms de produits qui doivent obligatoirement avoir la même syntaxe que les produits de la feuille Products. Pour permettre un produit sur des flux, il suffit d'ajouter un « x » dans la colonne du produit sur chaque ligne des flux que vous désirez. Alternativement, vous pouvez ajouter une seule colonne nommée « AUTO » et le mapping des produits/flux se construira automatiquement selon les entrées et les sorties des différents BU.

2.2 Procédure de génération du fichier d'entrée Access

Cette section montre les étapes à suivre pour générer un fichier d'entrée LogiLab à partir d'un fichier de saisi Excel.

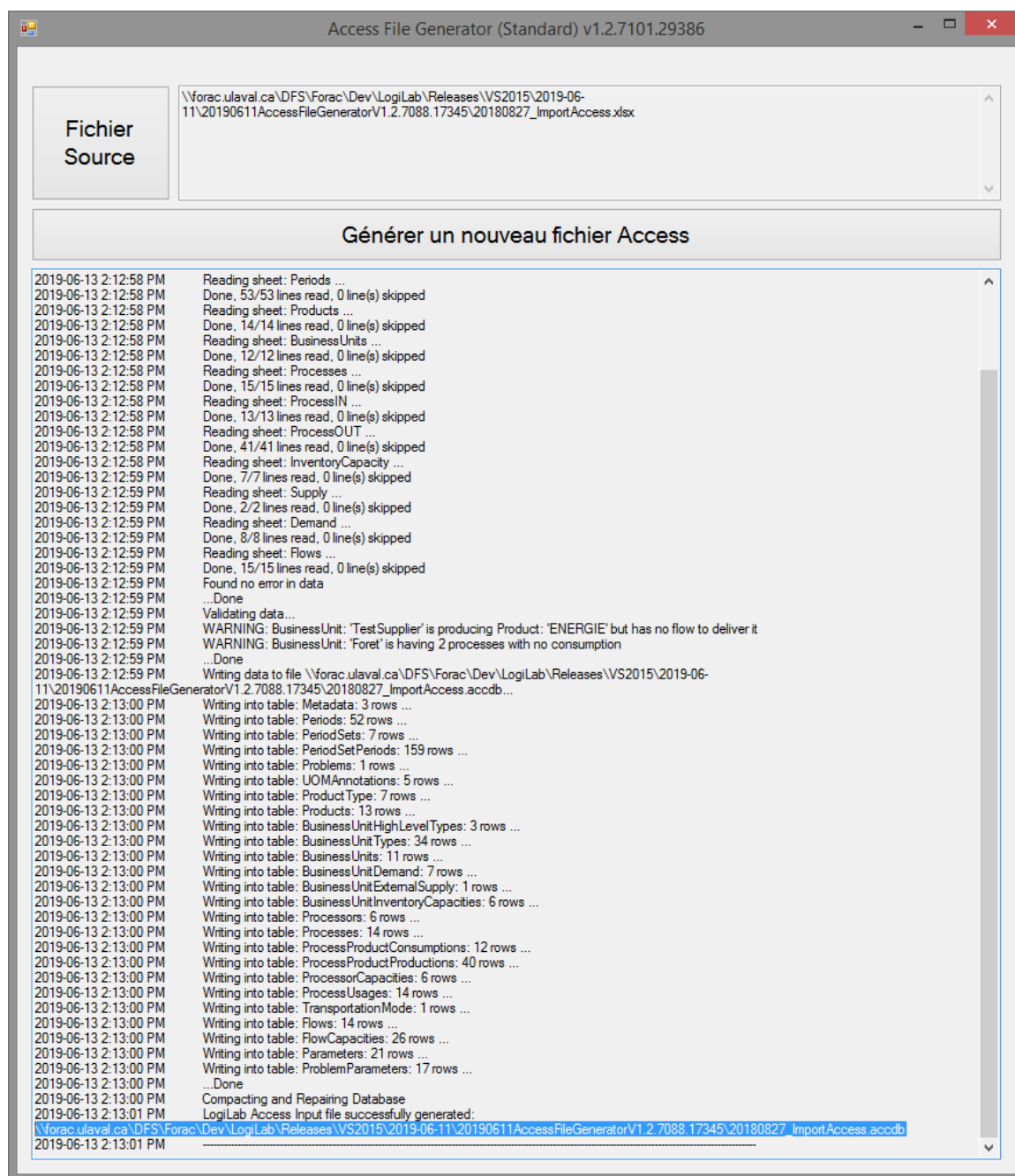
- 1- Assurez-vous d'avoir bien saisi toutes les informations du fichier de saisi Excel tel que décrit dans la section 2.1.
- 2- Dans le répertoire de l'application téléchargé, exécuter l'application « Forac.LogiLab.AccessFileGenerator.exe ». Cela ouvrira la fenêtre suivante :



- 3- Cliquez sur le bouton « Fichier Source » et sélectionnez votre fichier Excel de saisi. Le fichier sélectionné s'affichera dans la boîte de texte du haut tel qu'illustré ci-dessous :



- 4- Appuyez sur le bouton « Générer un nouveau fichier Access » :



- 5- La boîte de texte du bas se remplit avec le journal de conversion du fichier. Il est recommandé de bien lire ce journal pour valider que toutes les lignes de votre fichier Excel de saisi ont été lus, que la validation des données ne vous indique pas d'erreur ou d'avertissement anormal. Finalement, l'emplacement de du fichier Access généré est indiqué à la fin du journal.
- 6- Vous pouvez maintenant importer votre fichier Access dans LogiLab en suivant la procédure standard.

2.3 Résolution de problèmes

L'objectif de cette section est d'aider les usagers à corriger eux-mêmes les problèmes rencontrés avec l'application.

2.3.1 Could not write into Access file, please install 'Microsoft Access Database Engine 2010 Redistributable' for X64 architecture

Cette erreur survient parce que votre machine ne respecte pas les prérequis de l'application. Suivez la procédure d'installation d'un « Access Redistributable x64 » tel que décrit à la section 2.0.