

AFile

Un gestionnaire de fichiers

Copyright © 1993-1994 par Denis GOUNELLE.

Toute utilisation commerciale ou vente sans autorisation écrite est strictement interdite. Vous pouvez copier et diffuser ce programme aux conditions suivantes :

1. L'ensemble des fichiers doit être fourni
2. Aucun fichier ne doit avoir été modifié
3. Vous ne devez pas demander plus de 40FF pour cela

Malgré de nombreux tests, je ne peux garantir que AFile ne contient aucune erreur. VOUS UTILISEZ CE PROGRAMME A VOS RISQUES ET PERILS. Je ne pourrai en aucun cas être tenu pour responsable de tout dommage, direct ou indirect, résultant de l'utilisation de AFile.

1 Introduction

AFile est gestionnaire de fichier, c'est-à-dire qu'il vous permet de créer et de gérer vos fichiers (carnet d'adresse, vidéothèque, clients, etc...) au moyen d'une interface Intuition. Il n'y a aucune limitation sur le nombre de champs ou d'enregistrements. Le programme dispose des fonctions classiques d'impression, de tri et d'importation/exportation de données.

AFile utilise AREXX comme langage de programmation. Il est possible de créer des masques de saisie complets avec image de fond, positionnement des champs au pixel près, vérification des valeurs saisies, personnalisation des menus et de l'impression.

Vos critiques et suggestions sur ce programme seront toujours les bienvenues. N'hésitez pas à m'écrire, à l'adresse suivante :

M. GOUNELLE Denis
27, rue Jules GUESDE
45400 FLEURY-LES-AUBRAIS
FRANCE

Vous pouvez également m'adresser un message à l'adresse Internet "gounelle@alphanet.ch". Notez qu'il s'agit d'une boîte à lettre mise à ma disposition par un ami, aussi veuillez n'envoyer que de courts messages. De plus, n'ayant pas accès directement à mes messages, n'espérez pas avoir une réponse avant une dizaine de jours.

Merci à Yves PERRENOUD pour ses nombreuses suggestions.

2 Installation

AFile a besoin de la bibliothèque `'mathieeedoubbas.library'` qui est fournie par Commodore sur les disquettes système.

AFile est désormais localisé, c'est-à-dire qu'il peut s'adapter à la langue par défaut si vous avez le système 2.1 ou plus. Il vous faudra alors copier le fichier catalogue désiré dans le répertoire correspondant à votre langue par défaut. Par exemple, s'il s'agit du français, copiez le fichier `'français.catalog'` dans le répertoire `'SYS:Locale/Catalogs/Français'`, sous le nom `'AFile.catalog'`.

3 Démarrage

Le programme peut être appelé du CLI ou du Workbench. Dans les deux cas, il est possible d'indiquer le nom du fichier à utiliser, par la méthode habituelle, ce qui permet d'accéder directement à l'écran de visualisation/modification des données.

De plus, les arguments suivants peuvent être indiqués :

USEASL permet de demander à utiliser la requête de fichier de 'as1.library' au lieu de la requête interne. Cet argument est ignoré sous Kickstart 1.3.

NOCASE l'option "minuscules = MAJUSCULES" est sélectionnée par défaut pour le tri et la recherche (depuis l'interface Intuition seulement)

PRINTSCRIPT <nom de script>
PRINTSCRIPT=<nom de script>

permet d'indiquer le script AREXX à utiliser pour l'impression des enregistrements, en remplacement de l'impression par défaut.

Le script sera appelé pour chaque enregistrement à imprimer, avec le contenu de cet enregistrement directement accessible par la commande *GETFIELD*. La sortie standard est automatiquement redirigée vers l'imprimante : vous n'aurez donc qu'à utiliser la commande "SAY" pour envoyer les données. Voir Chapitre 11 [Interface AREXX], Page 20.

Vous pouvez également spécifier le nom du script d'impression dans les masques de saisie. Voir Chapitre 9 [Masques de saisie], Page 14.

INPUTMASK <nom du masque>
INPUTMASK=<nom du masque>

permet d'indiquer un masque de saisie dès le démarrage. Cette option est surtout utile lorsqu'un nom de fichier est également indiqué.

FONT <nom>
FONT=<nom>

permet d'indiquer la police à utiliser, au lieu de la police texte par défaut. Le nom de la police doit être sous la forme <nom de la police><taille en Y> (par exemple "courier9"). Note que AFile ne peut utiliser une police proportionnelle.

DEPTH <nombre de plans>
DEPTH=<nombre de plans>

permet d'indiquer le nombre de plans que doit avoir l'écran de travail. Cette option est utile lorsqu'un masque de saisie charge une image de fond et que vous voulez être certain d'avoir le bon nombre de couleurs.

AREXXWIN <spécification de fenêtre>

AREXXWIN=<spécification de fenêtre>

permet d'indiquer une spécification de fenêtre pour l'affichage des résultats des scripts AREXX lancés depuis AFile. Par défaut, AFile ouvre une fenêtre couvrant tout l'écran, sauf la barre de menus.

4 Menu principal

Ce menu s'affiche automatiquement au démarrage du programme, si vous n'avez pas indiqué de nom de fichier comme argument. AFile utilise automatiquement la police texte et la taille d'écran par défaut. Le menu offre les fonctions suivantes :

Nouveau fichier

Permet de créer un nouveau fichier. Une requête de fichier vous permettra d'indiquer le nom du fichier à créer. Voir Chapitre 5 [Définition de structure], Page 6.

Modifier structure

Permet de modifier la structure d'un fichier existant. Une requête de fichier vous permettra de choisir le fichier à modifier. Voir Chapitre 6 [Modification de structure], Page 8.

Ouvrir fichier

Permet d'ouvrir un fichier existant. Une requête de fichier vous permettra de choisir le nom du fichier à ouvrir, puis l'écran de visualisation/modification des données apparaîtra. Voir Chapitre 7 [Accès aux données], Page 9.

Exécuter script

Permet d'exécuter un script AREXX. Une requête de fichier vous permettra de choisir le script à exécuter. Voir Chapitre 11 [Interface AREXX], Page 20.

Quitter

Termine l'exécution de AFile.

5 Définition de structure

AFile gère des fichiers composés d'enregistrements de longueur fixe. Tous les enregistrements ont la même structure, composée d'un nombre illimité de champs définis par leur nom, leur taille et leur type. Le nombre d'enregistrements par fichier n'est limité que par la capacité de votre disque dur.

Les noms des champs ont une longueur maximale de 32 caractères, et peuvent contenir n'importe quel caractère. Il est cependant conseillé de ne pas utiliser d'espaces si vous prévoyez un accès depuis AREXX. Il ne peut y avoir deux champs de même nom dans la structure d'un fichier.

La taille des champs est limitée en théorie à 65535 caractères, sauf pour les champs de type "DATE", "BOOLEEN" et "ENUM", qui ont une taille de 8, 1 et 1 octets respectivement. Pour les champs numériques, vous pouvez également définir le nombre de chiffres après la virgule.

Les types de champs sont les suivants :

<i>alpha</i>	valeur alphabétique uniquement (lettres majuscules et minuscules, avec ou sans accents, espace, tiret, apostrophe)
<i>numeric</i>	valeur numérique avec ou sans virgule
<i>alphanum</i>	valeur quelconque
<i>date</i>	date (jour, mois, année avec siècle)
<i>booleen</i>	valeur booléenne (VRAI ou FAUX)
<i>enum</i>	valeur parmi une liste indiquée lors de la définition

La requête de définition de structure se compose d'une zone d'affichage où apparaissent les définitions des champs (nom/type/longueur). Un ascenseur permet de faire défiler l'affichage. Le champ courant est affiché sur un fond bleu. Pour sélectionner un champ, il suffit de cliquer sur sa définition : elle sera automatiquement recopiée dans les gadgets.

Les gadgets en bas de la fenêtre effectuent les actions suivantes :

<i>Ajoute</i>	ajoute le nouveau champ comme dernier champ
<i>Insère</i>	insère le nouveau champ avant le champ courant
<i>Supprime</i>	supprime le champ courant

Remplace remplace le champ courant par un nouveau champ. Ceci permet par exemple de renommer un champ, ou de changer sa taille.

Pour définir un nouveau champ, entrez ses caractéristiques dans les gadgets de chaîne, et sélectionnez son type avec le gadget *Type*. Pour les champs de type "ENUM", il vous faudra indiquer la liste des valeurs possibles, séparés par des virgules, dans le gadget *Valeurs*. Cliquez ensuite sur le gadget *Ajoute* : le champ apparaît alors dans la zone d'affichage, et devient le champ courant. Si l'ajout est refusé (il y a déjà un champ avec le même nom, plus de mémoire, etc...), l'écran flashe.

La requête de définition de structure possède également deux menus, qui proposent les fonctions suivantes :

Copie structure

Permet de copier la structure d'un fichier existant à la place de la structure actuellement affichée dans la requête. Une requête de fichier vous permettra d'indiquer le fichier dont vous voulez copier la structure.

Abandonne

Permet d'abandonner la définition de structure et de revenir au menu principal, sans créer de fichier.

Enregistre Permet d'enregistrer la définition de structure, et de créer physiquement le fichier. AFile revient ensuite au menu principal.

Crypte les données

Les données du fichier seront cryptées à l'aide d'un mot de passe, qui vous sera demandé lorsque vous enregistrerez la définition de structure. Ce mot de passe devra ensuite être donné par l'utilisateur chaque fois qu'il voudra accéder au fichier.

Attention, si vous oubliez le mot de passe il vous sera impossible de récupérer les données du fichier !

6 Modification de structure

La modification de structure vous permet de modifier la structure d'un fichier existant, c'est-à-dire d'ajouter des champs, d'en supprimer, ou d'en modifier. Notez qu'il est impossible de modifier le type d'un champ, et de détruire tous les champs. A part cela, toutes les autres caractéristiques peuvent être changées.

La requête de modification est la même que celle utilisée pour la définition du fichier. Voir Chapitre 5 [Définition de structure], Page 6. Les seules différences sont que le gadget *Type* et l'item *Copie struct.* sont désactivés.

Une fois la structure modifiée, utilisez l'item *Enregistre* pour valider vos modifications : AFile convertira alors automatiquement le fichier dans la nouvelle structure. Notez que les enregistrements marqués comme détruits sont perdus lors de cette opération, et que cette conversion peut échouer s'il ne reste pas assez de place sur le volume où se trouve le fichier.

Si la conversion fait que des données seraient perdues (champs supprimés, champs plus courts, ou enregistrements marqués comme détruits), AFile vous demandera de confirmer l'opération.

A l'aide du menu *Options*, vous pouvez supprimer ou ajouter le cryptage des données. Si l'option de cryptage est activée lorsque vous enregistrerez la nouvelle structure, AFile vous demandera un mot de passe. Notez que cela vous permet de changer le mot de passe protégeant un fichier : il vous suffit de demander une modification de structure du fichier, d'enregistrer la structure sans aucune modification et d'entrer le nouveau mot de passe.

7 Accès aux données

L'écran se compose d'une zone principale dans laquelle sont affichées les données, d'une ligne d'état et une rangée de gadgets.

Les gadgets contenant les données sont en principe de la même largeur que le champ correspondant, sauf si l'écran n'est pas assez large. Lorsque vous modifiez la valeur d'un champ, AFile vérifie que la nouvelle valeur est compatible avec le type du champ. Les dates doivent être introduites sous la forme "JJ/MM/AAAA" (avec siècle), ou simplement "JJ/MM/AA" (le siècle courant est automatiquement ajouté). La spécification "?" permet d'introduire facilement la date du jour (voir Chapitre 10 [Alias], Page 19).

La première ligne d'état indique le type et la taille du champ courant. La seconde ligne indique le nom du fichier et la position courante sous la forme X/Y : X est le numéro de l'enregistrement courant, Y le numéro du dernier enregistrement. Lorsqu'un tri est actif, le nom du fichier est suivi de la mention (*trié*). Si l'enregistrement courant a été modifié, la mention *Modifié* s'affiche à droite.

La rangée de gadget permet les opérations suivantes (de gauche à droite) :

- accès au premier enregistrement (touche **shift-flèche gauche**)
- accès à l'enregistrement précédent (touche **flèche gauche**)
- écriture de l'enregistrement courant
- accès à l'enregistrement suivant (touche **flèche droite**)
- accès au dernier enregistrement (touche **shift-flèche droite**)
- accès à l'enregistrement n (touche **g**)
- effacement de l'enregistrement courant (touche **DEL**)
- impression de n enregistrements à partir de l'enregistrement courant (touche **p**)
- page des champs précédente (touche **flèche haut**)
- page des champs suivante (touche **flèche bas**)

Si l'enregistrement courant est modifié, ces modifications seront automatiquement écrites si vous changez d'enregistrement. Si vous êtes positionné sur le dernier enregistrement, demander à accéder à l'enregistrement suivant créera automatiquement un nouvel enregistrement.

Les autres opérations sont accessibles via trois menus.

7.1 Le menu Fichier

Récupère enreg.

Pour AFile, effacer un enregistrement signifie positionner un indicateur au début de l'enregistrement, qui signale que cet enregistrement est effacé, et qu'il ne faut plus y accéder. L'enregistrement n'est donc pas détruit physiquement. Cette option du menu vous permet de récupérer tous les enregistrements marqués comme détruits, pour pouvoir y accéder de nouveau.

Compacte fichier

Permet de détruire physiquement les enregistrements marqués comme détruits. Cela veut dire que ces enregistrements ne pourront plus être récupérés. En contrepartie, la taille du fichier diminuera, car les enregistrements détruits seront éliminés. AFile vous demandera confirmation avant de lancer l'opération. Notez que cette opération peut échouer s'il ne reste pas assez de place sur le volume où se trouve le fichier.

Informations

Affiche des informations sur le fichier courant : son nom, le nombre de champs, la taille d'un enregistrement, le nombre d'enregistrements, et le nombre d'enregistrements marqués comme détruits.

Imprime structure

Imprime la structure du fichier, c'est-à-dire les définitions des champs et la taille des enregistrements. L'impression peut être envoyée vers un fichier.

Ferme fichier

Permet de fermer le fichier et de revenir à l'écran principal.

Fermer et quitter

Permet de fermer le fichier et quitter le programme.

7.2 Le menu Edition

Annule modifs

Permet de relire l'enregistrement courant, ce qui annule toute modification des données dans les champs (à condition que ces modifications n'aient pas été déjà enregistrées).

Select. masque

Sélectionne un masque de saisie. Une requête de fichier apparaîtra pour voir permettre de choisir le masque. Voir Chapitre 9 [Masques de saisies], Page 14.

Sans masque

Désactive le masque de saisie courant.

Importer Permet d'importer des données, c'est-à-dire de récupérer des données provenant d'un fichier ASCII. Les nouveaux enregistrements seront ajoutés à la fin du fichier. Voir Chapitre 8 [Importation/Exportation], Page 13.

Exporter Permet d'exporter des données, c'est-à-dire d'extraire des données vers un fichier ASCII. Tous les enregistrements du fichier seront exportés. Voir Chapitre 8 [Importation/Exportation], Page 13.

Chercher Permet de chercher un enregistrement contenant une valeur précise dans un champ. Une boîte de requête vous permettra de sélectionner le champ sur lequel vous voulez faire une recherche. Vous devez ensuite indiquer la chaîne à rechercher, et les options de recherche :

Exactement

Cherche un champ ayant exactement la valeur indiquée (option activée) ou contenant la valeur (option désactivée)

min = MAJ

Ne différencie pas minuscules et majuscules (option activée) ou les différencie (option désactivée)

Différent

Recherche inversée (option activée, AFile s'arrêtera sur le premier enregistrement dont le champ ne correspond pas) ou recherche normale (option désactivée)

Les deux dernières options ne sont utilisables que pour des champs de types "ALPHA" ou "ALPHANUM".

Chercher suivant

Permet de poursuivre une recherche.

7.3 Le menu Spécial

Ajouter critère tri

Permet d'ajouter un critère de tri supplémentaire. Une première boîte de requête vous permettra d'indiquer le champ sur lequel effectuer le tri. Une seconde boîte de requête vous permettra d'indiquer les paramètres du tri : ordre croissant ou décroissant, avec ou sans différenciation des minuscules et des majuscules. Cette dernière option n'est utilisable que pour les champs de types "ALPHA" ou "ALPHANUM".

Une fois le tri effectué, AFile se positionne sur le premier enregistrement dans l'ordre du tri. Tous les déplacements (premier/précédent/suivant/dernier) se font ensuite en utilisant l'ordre du tri. Notez que AFile réalise un tri logique, et non physique : l'ordre des enregistrements dans le fichier n'est pas modifié. Un tri peut réclamer beaucoup de mémoire si le fichier contient beaucoup d'enregistrements. Tant qu'un tri

est actif, les opérations *Récup. enreg* et *Compacte fichier*, l'ajout ou la suppression d'enregistrements, ainsi que la modification du champ de tri sont interdits.

Enlever critère tri

Permet d'enlever soit le dernier, soit tous les critères de tri. Une fois que tous les critères de tri ont été enlevés, l'accès aux enregistrements se fait de nouveau dans l'ordre physique dans le fichier, et il est possible d'ajouter ou supprimer des enregistrements.

Reclasser fichier

Modifie l'ordre physique des enregistrements, pour qu'il soit conforme au tri actif. Notez que les enregistrements marqués comme détruits sont perdus lors de cette opération, et que ce reclassement peut échouer s'il ne reste pas assez de place sur le volume où se trouve le fichier. AFile vous demandera confirmation avant de lancer l'opération.

Exécuter script

Permet d'exécuter un script AREXX. Une requête de fichier vous permettra de choisir le script à exécuter. Voir Chapitre 11 [Interface AREXX], Page 20.

8 Importation/Exportation

La sélection de l'item *Importer* (ou de l'item *Exporter*) fait apparaître un boîte de requête, permettant d'indiquer les paramètres nécessaires à l'importation (ou à l'exportation) des données.

Le premier paramètre est le nom du fichier ASCII source (ou cible).

Il vous faudra ensuite indiquer le caractère séparateur de champs, et le caractère séparateur d'enregistrements. Ces caractères peuvent être indiqués directement (par exemple "/"), ou sous forme de code ASCII décimal (par exemple "10" pour une fin de ligne). Notez que dans ce cas, le code ASCII doit faire au moins deux chiffres : "2" correspond au caractère "2", "02" correspond au caractère de code ASCII égal à 2.

Il vous faudra enfin indiquer un numéro d'ordre pour chaque champ : les champs seront importés (ou exportés) dans l'ordre indiqué. Les champs pour lesquels vous n'aurez rien indiqué ne seront pas importés (ou exportés). Lors de l'importation, le numéro d'ordre peut être assimilé à un numéro de colonne dans le fichier source. Il permet donc d'ignorer des données, ou d'intervertir l'ordre des données.

9 Masques de saisie

Les masques de saisie permettent de réaliser de nombreuses fonctions : personnaliser l'affichage des données, assister l'utilisateur pendant la saisie, vérifier les données saisies, ajouter un menu personnalisé, personnaliser l'impression.

Pour AFile, un masque se présente sous la forme d'un script AREXX contenant des commentaires dans un format particulier. Le format général de ce script est le suivant :

```

/*
 * Exemple de masque de saisie pour AFile
 *
 * $MSG
 *     <spécification de message>
 * $END
 *
 * $MASK <image de fond>
 *     <spécification de masque>
 * $END
 *
 * $MENU
 *     <spécification de menu>
 * $END
 *
 * $PRINT <script d'impression>
 *
 * $PRINTFMT <format d'impression>
 */

PARSE ARG champ valeur

<instructions AREXX>

EXIT 0

```

Il n'est pas obligatoire d'indiquer toutes les spécifications : le masque peut par exemple ne contenir qu'une spécification \$MSG. A l'intérieur d'une spécification, il est possible d'indiquer des commentaires de la façon suivante :

```

/* $MSG
 *     ligne de spécification
 **     ligne de commentaire (commence par " **")
 *     ligne de spécification
 * $END
 */

```

La sélection d'un masque se fait à l'aide de l'item de menu *Select. masque*. Pour annuler la sélection d'un masque, utilisez l'item *Sans masque*. Voir Paragraphe 7.2 [Le menu Edition], Page 10.

Enfin, il est possible d'associer un masque à un fichier, de façon à ce qu'il soit automatiquement chargé à l'ouverture de ce fichier. Il suffit que le fichier masque ait pour nom le nom du fichier de données auquel il se rapporte, suivi du suffixe `.mask`. Par exemple, si votre fichier de données s'appelle `Adresses`, le nom du fichier masque devra être `Adresses.mask`. Les deux fichiers doivent se trouver dans le même répertoire.

9.1 Spécification \$MSG

Les lignes de spécification doivent indiquer :

1. un nom de champ
2. s'il faut vérifier la valeur saisie
3. un message

Voici un exemple :

```
/* $MSG
**      Nom du champ      Vérification?      Libellé
*      NUMERO             -                      Numéro de produit
*      QUANTITE           CHECK                   Quantité arrivée
*      LIVREUR            -                      Nom du livreur
*      DATE_ENTREE       CHECK                   Date d'arrivée
* $END
*/
```

Le message est affiché au dessus de la ligne d'état lorsque le champ est sélectionné. Si le mot CHECK figure dans la seconde colonne, le script AREXX contenant le masque de saisie est appelé à chaque modification du champ, avec comme arguments le nom du champ et la nouvelle valeur. Il doit renvoyer un code retour nul si cette valeur est acceptée, un code retour non nul si cette valeur est refusée. Les valeurs de type DATE seront passées sous la forme "AAAAMMJJ", et les valeurs booléennes seront égales soit à "T", soit à "F".

Pendant l'exécution du script, les ordres SEEK, WRITE, APPEND, DELETE, SEARCH, SORT, RELEASE, SETFIELD, et CLOSE sont interdits sur le fichier courant. Voir Chapitre 11 [Interface AREXX], Page 20.

9.2 Spécification \$MASK

Cette spécification est plus complète que la spécification \$MSG, car elle permet de charger une image de fond, et de positionner les champs sur l'écran. Les lignes de spécification doivent indiquer :

1. un nom de champ
2. une position en X (en pixels)
3. une position en Y (en pixels)
4. une largeur de champ (en pixels)
5. s'il faut vérifier la valeur saisie
6. un message

Voici un exemple :

```
* $MASK          Work:Images/AFile/Adresse.pic
**      Nom du champ   PosX   PosY   Largeur Vérif?  Message
*      NUMERO         400    4     -      -      Numéro de client
*      PRENOM        400    26    100    -      Prénom du client
*      NOM           400    48    100    -      Nom du client
* $END
*/
```

Le nom de fichier (optionnel) après le mot-clé \$MASK doit correspondre à un fichier image, au format IFF. Cette image est affichée en image de fond, lors de la prise en compte du masque de saisie. La palette de l'image est chargée également.

AFile est capable de charger une image qui est dans une résolution différente de celle de l'écran qu'il utilise. Vous pouvez utiliser l'argument DEPTH pour augmenter le nombre de couleurs (voir Chapitre 3 [Démarrage], Page 3).

Les positions en X et en Y, ainsi que la largeur, sont utilisés pour positionner les champs sur l'écran. Si la largeur indiquée est "-", la largeur par défaut est utilisée.

Le message est affiché au dessus de la ligne d'état lorsque le champ est sélectionné. Si le mot CHECK figure dans la cinquième colonne, le script AREXX contenant le masque de saisie est appelé à chaque modification du champ, avec comme arguments le nom du champ et la nouvelle valeur. Il doit renvoyer un code retour nul si cette valeur est acceptée, un code retour non nul si

cette valeur est refusée. Les valeurs de type DATE seront passées sous la forme "AAAAMMJJ", et les valeurs booléennes seront égales soit à "T", soit à "F".

Pendant l'exécution du script, les ordres SEEK, WRITE, APPEND, DELETE, SEARCH, SORT, RELEASE, SETFIELD, et CLOSE sont interdits sur le fichier courant. Voir Chapitre 11 [Interface AREXX], Page 20.

9.3 Spécification \$MENU

Les lignes de spécification doivent indiquer :

1. le titre de l'item du menu
2. s'il faut demander des arguments
3. le nom du script AREXX à appeler

Voici un exemple :

```
/* $MENU
**      Nom de l'item      Arguments?      Nom du script
*      Statistiques      -              Work:Lib/Stats.rexx
*      MonExport          ARGS,QUIET      Work:Lib/Export.rexx
* $END
*/
```

Les items seront ajoutées dans un nouveau menu : le menu *Extras*. Les dix premiers items disposeront de raccourcis claviers. Chaque fois qu'un des items sera sélectionné, AFile appellera le script AREXX correspondant, après avoir demandé des arguments si le mot ARGV figure dans la seconde colonne. Le script sera exécuté sans fenêtre de sortie si le mot QUIET figure dans la seconde colonne.

9.4 Spécification \$PRINT

Le mot-clé \$PRINT doit être suivi du nom du script AREXX à utiliser pour imprimer les enregistrements.

Le script sera appelé pour chaque enregistrement à imprimer, avec le contenu de cet enregistrement directement accessible par la commande AREXX *GETFIELD*. La sortie standard est

automatiquement redirigée vers l'imprimante : vous n'aurez donc qu'à utiliser la commande "SAY" pour envoyer les données. Voir Chapitre 11 [Interface AREXX], Page 20.

Cette spécification remplace le script éventuellement indiqué par l'argument PRINTSCRIPT (voir Chapitre 3 [Démarrage], Page 3), tant que le masque de saisie est actif. Toute spécification \$PRINTFMT indiquée avant une spécification \$PRINT sera ignorée.

9.5 Spécification \$PRINTFMT

Le mot-clé \$PRINTFMT doit être suivi d'une chaîne format, indiquant comment imprimer les enregistrements. Pour chaque enregistrement à imprimer, les données seront mises en forme à l'aide de cette chaîne format, avant envoi vers l'imprimante.

La chaîne de format peut contenir les spécifications suivantes :

`%no[:{l|c|r}[taille]]`

Remplacé par le contenu du champ numéro "no", éventuellement justifié à gauche (l), centré (c) ou justifié à droite (r) sur "taille" caractères. Si "taille" est omis, la taille du champ est utilisée. Exemples de spécifications : `%3 %2:r %11:c20`

`$no[:{l|c|r}[taille]]`

Remplacé par le nom du champ numéro "no", éventuellement justifié à gauche (l), centré (c) ou justifié à droite (r) sur "taille" caractères. Si "taille" est omis, la taille du champ est utilisée.

`\n` Remplacé par un saut de ligne

`\f` Remplacé par un saut de page

`\caractère` Remplacé par le caractère indiqué. Ceci permet d'inclure les caractères spéciaux suivants : %, \$, \

Toute spécification \$PRINT indiquée avant une spécification \$PRINTFMT sera ignorée.

10 Les Alias

Un alias est un moyen de faciliter l'entrée des données, en permettant de définir des abbréviations. Un exemple est l'introduction de "?" dans un champ de type DATE, pour entrer la date du jour. Cette possibilité est intégrée dans AFile (voir Chapitre 7 [Accès aux données], Page 9), mais vous pouvez également définir vos propres alias, à l'aide d'un fichier texte composé d'une suite de lignes indiquant :

1. le nom du champ
2. la valeur de l'alias
3. la valeur correspondante

Voici un exemple :

```
;Champ  Alias  Valeur
PAYS    RFA      REPUBLIQUE FEDERALE D'ALLEMAGNE
PAYS    USA      ETATS-UNIS D'AMERIQUE
PAYS    UK       ROYAUME-UNI
```

Avec ces définitions, chaque fois que l'utilisateur entrera la valeur "USA" dans le champ "PAYS", cette valeur sera automatiquement remplacée par la valeur "ETATS-UNIS D'AMERIQUE". Notez que le remplacement des alias se fait avant l'exécution d'un éventuel script AREXX de vérification de saisie (voir Chapitre 9 [Masques de saisie], Page 14).

Le fichier d'alias doit avoir pour nom le nom du fichier de données auquel il se rapporte, suivi du suffixe '.alias'. Par exemple, si votre fichier de données s'appelle 'Adresses', le nom du fichier d'alias devra être 'Adresses.alias'. Les deux fichiers doivent se trouver dans le même répertoire. Le fichier d'alias est chargé automatiquement à l'ouverture du fichier de données.

11 Interface AREXX

AFile est entièrement interfacé avec AREXX, qui devient ainsi son langage de programmation. Le nom du port AREXX est 'AFile_rexx'. Il est possible de lancer un script par l'item *Exécute AREXX*, soit depuis le menu *Projet* de l'écran principal, soit depuis le menu *Spécial* de l'écran d'accès aux données. Dans ce dernier cas, le fichier courant est automatiquement sélectionné comme fichier de travail, et la position dans le fichier est celle de l'enregistrement actuellement affiché.

Voici la liste des commandes reconnues :

OPEN nom

Ouvre le fichier indiqué, et se positionne sur le premier enregistrement.

Ce fichier **n'est pas** sélectionné comme fichier de travail.

SELECT nom

Sélectionne le fichier indiqué comme fichier de travail. Ce fichier doit avoir été ouvert par une commande *OPEN*.

CLOSE nom

Ferme le fichier indiqué

CLOSE ALL

Ferme tous les fichiers ouverts

Les commandes suivantes sont appliquées au fichier de travail :

APPEND Crée un nouvel enregistrement, et se positionne dessus

SEEK FIRST

Se positionne sur le premier enregistrement

SEEK LAST

Se positionne sur le dernier enregistrement

SEEK NEXT

Se positionne sur l'enregistrement suivant

SEEK PREV

Se positionne sur l'enregistrement précédent

SEEK n Se positionne sur le n-ième enregistrement

SEARCH nom val [NOCASE] [EXACT] [NOTEQUAL]

Cherche un enregistrement dont le champ "nom" contient la valeur "val". L'option "NOCASE" permet une recherche sans différencier minuscules et majuscules. L'option

"EXACT" permet de chercher un champ qui est contient exactement la valeur indiquée. L'option "NOTEQUAL" permet une recherche inversée.

SEARCH NEXT

Poursuit la recherche

INFO Retourne des informations sur le fichier courant dans la variable RESULT, sous la forme :

"nom complet du fichier" <nombre de champs> <longueur des enregistrements> <nombre d'enregistrements> <nombre d'enregistrements détruits>

SORT nom [DECREASE] [NOCASE]

Ajoute le champ indiqué comme critère de tri supplémentaire. L'option "DECREASE" permet d'avoir un tri par ordre décroissant, l'option "NOCASE" permet de ne pas différencier minuscules et majuscules.

RELEASE {LAST|ALL}

Enlève le dernier, ou tous les critères de tri.

Les commandes suivantes sont appliquées à l'enregistrement courant du fichier de travail :

WRITE Écrit l'enregistrement courant

DELETE Détruit l'enregistrement courant

GETFIELD nom

Retourne le contenu du champ "nom" dans la variable RESULT

SETFIELD nom "val"

Modifie le contenu du champ "nom". **Attention, la valeur doit être mise entre guillemets !**

FIELDTYPE nom

Retourne le type du champ "nom", tel que déclaré dans la structure du fichier ("A", "N", "S", "D", "B", ou "E").

ENUMLIST nom

Retourne la liste des valeurs possibles pour le champ "nom". Les valeurs sont séparées par des virgules.

FIELDLEN nom

Retourne la longueur du champ "nom", telle que déclarée dans la structure du fichier. Pour les champs numérique, cette longueur tient compte du point décimal, s'il y a des décimales.

En cas d'échec, ces commandes retournent 10 dans la variable RC. Elles renvoient 0 si tout c'est bien passé.

12 Historique

AFile a été écrit en langage C, sur un Amiga 3000 UNIX-1 (10 Mo de mémoire, lecteur de disquettes interne HD, disques internes de 100 Mo et 160 Mo, lecteur SyQuest externe de 88 Mo, lecteur de disquettes externe, moniteur 1960 et imprimante Star LC24-10) connecté par un câble null-modem à un A500 avec 1 Mo de mémoire.

v1.00, 25-Jul-93, 60448 octets

Première version.

v1.10, 07-Aug-93, 62632 octets

Ajout des masques de saisie.

Ajout des commandes ARExx "FIELDTYPE" et "FIELDLEN".

Si un tri actif, indique "trié sur <nom champ>" au lieu de seulement "trié".

v1.20, 06-Sep-93, 64336 octets

Le nombre de champs n'est plus limité.

Gestion du tri modifié : sélection du champ par requête, option "min = MAJ", interface AREXX revue.

On peut sélectionner le fichier texte pour importer/exporter à l'aide d'une requête de fichier.

v1.21, 12-Sep-93, 64500 octets

Utilise les nouvelles versions des fonctions OuvreEcran(), RequeteFic() et GetDefault-TextFont().

v1.30, 15-Sep-93, 65612 octets

Entièrement localisé.

v1.31, 07-Oct-93, 65704 octets

En fin d'exécution d'un script AREXX, attend l'appui sur RETURN pour fermer la fenêtre.

Erreur corrigée: le tri sur les champs numérique se faisait dans l'ordre inverse de celui demandé par l'utilisateur.

v1.40, 14-Oct-93, 67036 octets

On peut modifier la structure d'un fichier existant.

Ajout des alias.

Signale les erreurs éventuelles lors du chargement des masques de saisie et des alias.

Erreur corrigée dans la gestion des requêtes de chaînes.

Un peu de ménage dans le source.

v1.41, 17-Nov-93, 67292 octets

Met à jour l'entête du fichier à chaque effacement/ajout d'enregistrement (au lieu d'attendre la fermeture).

Amélioration de plusieurs parties du source (macros pour manipuler le champ `dfL.Flags`, etc...).

Lorsqu'une requête est affichée, bloque la fenêtre principale avec un `Requester`.

Erreur corrigée: ne rafraichissait pas correctement les gadgets correspondant aux champs booléens.

Quelques modifications pour supporter le catalogue allemand.

v2.00, 28-Nov-93, 68960 octets

On peut maintenant crypter les fichiers.

Ajout de l'argument `"PRINTSCRIPT"`.

Erreur corrigée dans la copie rapide des définitions de champs: les champs `ln.Pred` du premier champ, et `ln.Succ` du dernier champ, n'étaient pas mis à jour.

v2.10, 11-Dec-93, 69508 octets

Modification du format des masques de saisie: il y a maintenant un seul fichier (le script `AREXX`) qui indique les messages, et fait la vérification. Les masques dans l'ancien format sont toujours acceptés.

Possibilité de charger automatiquement un masque de saisie à l'ouverture d'un fichier.

Les lignes de status et les gadgets sont maintenant positionnés sous le dernier gadget de champ, et non plus en bas de l'écran.

Dans la création/modification de structure, le changement du type ne modifie plus la longueur du champ quand on passe par `"DATE"` et `"BOOLEEN"`.

v2.20, 25-Dec-93, 70876 octets

Extension des masques de saisie: menu personnalisable avec `$MENU`, choix du script d'impression avec `$PRINT`.

Compilé avec le SAS/C 6.50.

v2.21, 26-Jan-94, 70876 octets

Erreur corrigée: ne restaurait pas le nom de l'écran public par défaut après l'exécution d'un script `AREXX`.

Indique ma nouvelle adresse.

v2.30, 20-Feb-94, 71412 octets

Gestion du clavier : touches du curseur (+ shift), "p", "g", "DEL".

Ajout de l'argument `USEASL`.

Une ligne commençant par " `**`" dans un masque est un commentaire.

Nombre d'enregistrements par défaut pour l'impression fixé à un.

Requête de champ sur fond bleu (au lieu du fond tramé).

Compilé avec le SAS/C 6.51.

v3.00, 26-Feb-94, 74696 octets

Ajout de la spécification \$MASK (masque complet avec image de fond et positionnement des champs sur l'écran).

Ajout de l'argument DEPTH.

Requête de fichier interne améliorée.

Gestion des erreurs améliorée (affiche des messages).

Documentation maintenant mise en page avec TexInfo/MakeGuide.

v3.01, 05-Mar-94, 74732 octets

Erreur corrigée: affichait un message d'erreur à l'ouverture d'un fichier sans fichier '.mask' associé.

Restaure la palette quand on ferme un fichier.

v3.02, 28-Mar-94, 75104 octets

Erreur corrigée : utilisait une fonction de la '`graphics.library`' V39 (donc le programme ne marchait que sous OS 3.0).

Erreur corrigée dans la gestion des messages AREXX (certaines commandes étaient rejetés)

Erreur corrigée : ne passait pas correctement les arguments au script AREXX lors de la vérification des valeurs.

v3.10, 06-Apr-94, 78960 octets

Ajout du type de champ "ENUM"

Fonction de recherche entièrement revue : ouvre une fenêtre de saisie d'options, ajout des options "recherche exacte" et "recherche inverse"

Ajout de l'argument "FONT"

Les champs de type "ALPHA" n'acceptent plus que les lettres majuscules et minuscules (avec ou sans accents) et l'espace, le tiret et l'apostrophe.

Affiche le type et la taille du champ courant dans la ligne d'état

Ajoute le chemin d'accès du fichier courant si un nom de script dans un masque est indiqué sans chemin

Pour les masques complets (spécification \$MASK), affiche les lignes d'état et les gadgets d'action tout en bas de l'écran

Requête de champs améliorée : si ajout/insertion/remplacement alors qu'il manque une donnée, automatiquement le gadget correspondant

Plusieurs erreurs corrigées (mangeait de la mémoire, problèmes de chargement des masques de saisie, etc...)

v3.11, 16-Apr-94, 79220 octets

Ajout de l'argument "NOCASE"

Réorganisation du source (interface accès aux données)

Verrouille les entrées sur la fenêtre de définition de structure quand la requête de fichier est affichée (option "copie de structure")

Suppression de la copie rapide des définitions des champs (impossible à faire "proprement", source de corruption mémoire)

Correction d'une erreur dans la lecture des arguments depuis le CLI

v3.12, 21-May-94, 79220 octets

L'interface gère maintenant les dates avec le siècle

Erreur corrigée : les "TOOL TYPES" n'étaient pas lus correctement

v3.13, 05-Jul-94, 79684 octets

Ajout de l'argument AREXXWIN

Le nom de l'écran public est "AFile"

Quelques erreurs corrigées dans le chargement des images IFF

v3.20, 27-Jul-94, 80368 octets

Modification du format des masques de saisies (ajout de la spécification \$PRINTFMT, des mots clés CHECK/ARGS/QUIET)

Ajout de l'option "Fermer et quitter" dans le menu "Fichier"

Erreur corrigée: dans certains cas ne réaffichait pas la barre des gadgets de déplacement

Erreur corrigée: après un tri ou compactage, "sans masque" produisait un affichage inutilisable, puis "select masque" faisait tout planter.

Erreur corrigée: si un masque ne contenait qu'une spécification \$PRINT, celle-ci était ignorée

v3.30, 25-Sep-94, 81060 octets

Le tri peut désormais se faire sur plusieurs critères.

Table des matières

1	Introduction	1
2	Installation	2
3	Démarrage	3
4	Menu principal	5
5	Définition de structure	6
6	Modification de structure	8
7	Accès aux données	9
	7.1 Le menu Fichier	10
	7.2 Le menu Edition	10
	7.3 Le menu Spécial	11
8	Importation/Exportation	13
9	Masques de saisie	14
	9.1 Spécification \$MSG	15
	9.2 Spécification \$MASK	16
	9.3 Spécification \$MENU	17
	9.4 Spécification \$PRINT	17
	9.5 Spécification \$PRINTFMT	18
10	Les Alias	19
11	Interface AREXX	20
12	Historique	22