

Fonctionnalités supplémentaires :

- Les dossiers privés sont accessibles par une seule personne (dans \$HOME)
chaque répertoire contient un espace de backup dans .abc

le répertoire depot dans \$HOME/biblio/depot un répertoire de dépôt est toujours public!!!

- Un tiroir public ou « classeur » ne contient que des dossiers non-personnels accessibles par plusieurs personnes, c'est-à-dire des groupes de travail ou le public en général.

Ces tiroirs (fichier .box) sont stockés dans des biblioPubs (?)

On peut s'y abonner (ils sont publics) en lecture et/ou écriture pour peu qu'on aie les droits (ACL) dessus.

- la variable \$BiblioDEPOT définie à l'ouverture du programme en fonction du groupe primaire de l'utilisateur désigne un répertoire dans lequel l'utilisateur a le droit de déposer des documents (w=où, r=non)
cette variable n'est peut-être pas utile.

l'accessibilité des dossiers est déterminé par les ACL ou permissions de partage (os)

- Mes documents pointe toujours vers \$HOME ou un de ses sous-répertoires. (set mesdocs=\$HOME)

- Infos disponibles à l'administrateur sur :

- les fichiers déposés dans depot (ex : copies d'élèves)
- les fichiers sauvegardés dans backup (ex : récupérer un fichier supprimé par erreur)
- les fichiers supprimés (qui et quand?)
- les fichiers copiés (qui et quand?)
- les fichiers synchronisés (quand?)

- (privé) : Les fichiers supprimés sont envoyés dans la corbeille (réseau) si elle existe sinon dans .abc

- (public) : Les fichiers supprimés sont envoyés dans .abc

- Seul l'administrateur a accès à .abc dans un dossier public

- Liste des documents du dossier

- nom du document (avec ou sans l'extension)

- taille, date, propriétaire (option)

- description du document

- sous-dossier annexe

- sort par type, alphabétique, numérique, date

- sous-dossiers interdits.

- clic_droit :

- * Ajouter un document

- * Envoyer vers...

- * Marquer comme important

- * Copier/déplacer

- * Renommer

- * Supprimer

- clic_droit sur onglet :

- * sort by

- * ajouter/modifier un dossier

- * sous-dossier annexe

- * actualiser

- Onglet par thème
 - un ou plusieurs mots-clés
 - dans le nom du document ou dans le contenu

Applications à un réseau d'établissement

BiblioRoot : du style GUNTDATA, c'est la base de l'arborescence pour les données utilisateur

\$HOME est sous BiblioRoot

chaque groupe (ex : une classe) possède un tiroir public dans lequel on trouve le dépôt

chaque membre du groupe est abonné automatiquement.

On peut aussi créer un tiroir public du style « cdi » auquel on s'abonne librement.

Analyse documentaire

- gérée par b_indexit

C'est la partie d'analyse documentaire de biblio. Ses objectifs sont :

- création de fiches comportant un descriptif rapide, un résumé (fiche de lecture), et une liste de mots-clés
- création d'un dictionnaire unique pour l'ensemble des documents
- création d'un index pointant les occurrences des mots du dictionnaire dans les documents analysés.

- croisement des listes de mots-clés ou d'expressions-clés pour déterminer des listes thématiques de documents (documents ayant le même thème)
- conseils d'organisation des documents en fonction des thèmes
- recherche d'un document par mot(s)-clé(s)

	<i>Classes d'installation</i>		
	<i>Serveur de réseau</i>	<i>Ordinateur autonome</i>	<i>Client de réseau</i>
définition	serveur de compte/de fichiers	une machine qui n'est pas reliée en permanence au réseau	terminal relié à un serveur de compte/de fichiers
contient	les dossiers perso des utilisateurs; perso home- groupes public biblio en cgi sous Apache /home en ftp et en smb OS = Linux 1 point de partage unique : BiblioRoot	les dossiers perso de l'utilisateur; Réplication totale du master; \$home et BiblioRoot en smb 2 points de partage : BiblioRoot ConfFolder (\$home)	Pas de réplication des perso (\$home); Réplication des contextes public;
synchronisation	pas de synchronisation	depuis et vers le master	depuis le serveur et vers le dépôt du serveur
master	oui	oui	non
teached	non	oui	oui

Répertoires utilisés/cr  s par biblio

<i>r��pertoire</i>	<i>utilit��</i>
(ex : c:\biblio)	on y cr��e les nouveaux dossiers publics
\$HOME	on y cr��e les fichiers de configuration, ainsi que les nouveaux dossiers priv��s
c:\program files\biblio ou /usr/X11R6/bin	les fichiers executables de biblio

Fichiers cr  s par biblio

Ces fichiers sont stock  s dans le r  pertoire perso de l'utilisateur, ou    d  faut (win98) dans le r  pertoire c:\biblio\« utilisateur ». les chemises persos sont stock  s idem. les chemises communes sont stock  es dans les BiblioPub;

Fichier - description - modifi�� par :		
biblio.ini	Configuration g��n��rale	biblio(RW)
« contexte ».lst	Liste format��e des fichiers du contexte, tiroir et chemise courants	b_filelist(W), b_clidroit(R), biblio(R)
\$USER.log	historique des op��rations mark delete copy	b_filelist(R), b_clidroit(W), b_synchronise(RW)
acces.log	historique des fichiers utilis��s.	biblio(W), b_config(R)
« contexte »- « tiroir ».box	Liste des chemises du tiroir	b_config(W), b_indexit(RW)
« contexte ».bmp	skin du contexte	b_config(W), biblio(R)
« contexte »-fr.idx	index de mots des documents du cntex	
« contexte »-fr.dic	dictionnaire des documents du cntex	

Les variables de biblio

Elle sont d  clar  es dans bib_utils

<i>variable</i>	<i>description</i>	<i>variable</i>	<i>description</i>
Contexte	d��finit un ensemble de tiroirs		
modeBiblio	<i>serveur, client, serveur-backup</i>		
modeMaster	<i>ftp</i> ou point de montage (<i>mountp</i>)	BiblioTrash	corbeille r��seau (w=all, r=root)
master	nom DNS ou IP du master	BiblioRessource	chemin vers les ressources (icones..)
		BiblioDepot	le dossier partag�� de la classe

biblio.ini

#contexte, mode serveur/client, master type, adress

Contexte actif

client, *serveur*, serveur-backup

type de network : *ftp* ou *mountp*

adress du ftp ou point de montage

#titre, racine des dossiers

message de titre

où créer les nouveaux dossiers?

#applications (sous forme logiciel/extension , sans espaces

traitement de texte/extension

tableur/extension

images/extension

photos/extension

audio/extension

video/extension

\$USER.log

#mark

liste des fichiers marqués comme importants

#copy

liste des fichiers copiés (ou déplacés) dans un autre dossier

#delete

liste des fichiers supprimés (ou déplacés)

#create

liste des fichiers créés

#synch_copy

liste des fichiers importés depuis le master

#synch_delete

liste des fichiers supprimés depuis le master

« contexte ».lst

tiroir actif
<chemises>
liste des chemises
</chemises>
chemise active
chemin complet de la chemise active
nombre de fichiers
taille totale
<fichiers>
fichier n°1 : Pretty Name
fichier n°1 : technical name
fichier n°1 : autres infos
fichier n°2 : Pretty Name
...
</fichiers>

« contexte »- « tiroir ».box

[titre du contexte] n°d'ordre du tiroir
Chemise n°1 : Technical Name
Chemise n°1 : Pretty Name
Chemise n°1 : Flag_sort bytype-byname-bynumber-bydate-extention?-description?-annexes?
Chemise n°2 : Technical Name
Chemise n°2 : Pretty Name
...
[Analyse du :] date en clair (datetime en Double)
[Chemise n°1 : Technical Name]
nom_de_fichier n°1#9mot-clé,mot-clé,...#9description rapide#9n°d'indexage(12bits)Auto/manuel(1bit)
descriptif du fichier n°1
nom_de_fichier n°2#9mot-clé,mot-clé,...#9description rapide#9n°d'indexage(12bits)Auto/manuel(1bit)
descriptif du fichier n°2
...
#end
#end

« tiroir ».dbx

fichier typé style dbf (sans entête?)

Entête :

Description des champs (longueur totale =) :

- nom du fichier : 64
- mot-clé,mot-clé,... : 32
- description rapide : 64
- description longue : 255
- n° d'index : 4
- Options : 1 (bit0=Auto/manuel)

Description des options :

bit0 = la description a été générée automatiquement ou bien entrée manuellement par un utilisateur.

Accès aux paramètres du logiciel

<p>Comment avoir la liste des contextes?</p>	<p><u>Perl</u> :</p> <pre>//lister les fichiers .box sortis par ordre alphabétique opendir(DIR, \$fromdir); @f=(?????????????) readdir DIR; closedir DIR; \$n=0; foreach \$f (@f) { //lire la chaine jusqu'à - et continuer si elle est déjà enregistrée next if ((???)=\$contexte[\$n]); \$n++; \$contexte[\$n]:=(????); }</pre> <p><u>Delphi</u> : (uses : classes; var : cntx : tstringlist; ts : tsearchrec)</p> <pre>//cntx : sorted & no_duplicates //lister les fichiers .box sortis par ordre alphabétique doserror:=FindFirst(filepath+'*.box',\$2D,ts); While doserror=0 do begin cntx.add(ts.Name); doserror:=findnext(ts); end; Sysutils.FindClose(ts);</pre>
<p>Comment avoir la liste des tiroirs du contexte?</p>	<p><u>Perl</u> :</p> <pre>//lister les fichiers contexte-???.box opendir(DIR, \$fromdir); @f=(/\$contexte-???.box) readdir DIR; closedir DIR; \$n=0; foreach \$f (@f) { //lire la chaine à partir de - \$n++; \$tiroir[\$n]:=(????); }</pre> <p><u>Delphi</u> : (uses : classes; var : tiroir : tstringlist; ts : tsearchrec)</p> <pre>//lister les fichiers .box sortis par ordre alphabétique doserror:=FindFirst(filepath+'contexte-*.box',\$2D,ts); While doserror=0 do begin tiroir.add(ts.Name); doserror:=findnext(ts); end; Sysutils.FindClose(ts);</pre>
<p>Comment déterminer le système d'exploitation?</p>	<p><u>Perl</u> :</p> <p><u>Delphi</u> : (uses : jclsysinfo; var winversion : twindowsversion;)</p> <p>GetWindowsVersion de JCLSysInfo</p>
<p>Comment déterminer le nom de l'utilisateur?</p>	<p><u>Perl</u> :</p> <p><u>Delphi</u> : (uses : jclsysinfo; var username : string;)</p> <p>GetLocalUserName de JCLSysInfo</p>

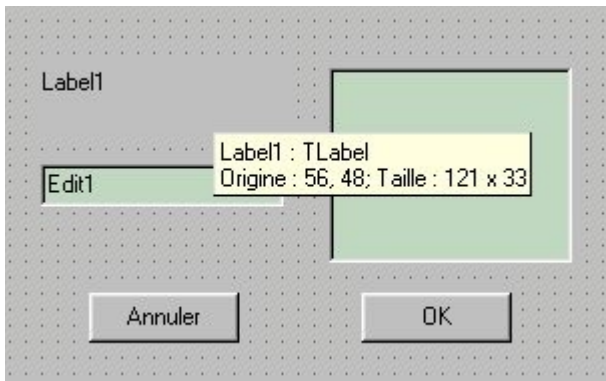
le programme b_configure (setup.exe)

Description :

```
# Usage : b_config [option]
#
# lancé sans option, le programme permet l'installation du logiciel
#
# options :
# - verify : vérifie la pertinence des fichiers de configuration.
# - config : permet à l'utilisateur de modifier la configuration
# - admin : permet de faire l'administration du système
#
#
#
# Installation :
# - Création des répertoires
# c:\biblio (biblioRoot) , c:\biblio\« utilisateur » si non $HOMEPATH (biblioConf),
# c:\program files\biblio (biblioBin)
# - Copier/créer les binaires et scripts et fichiers de conf
# binaires et scripts : biblio.exe, setup.exe, b_synchronise.pl, b_clic_droit.exe, b_filelist.exe,
# fichiers de conf : biblio.ini, « contexte ».lst, « contexte ».log, « contexte »- « tiroir ».box,
« contexte ».bmp
# - note : la création des fichiers de configuration doit être accompagnée d'un assistant
#
# Admin :
# - cherche les fichiers logs de tous les utilisateurs et fait une synthèse
#
#
#
#
```

le programme b_media

Description : c'est la commère du système, elle donne des infos sur les actions en cours; mais aussi prends en compte les choix de l'utilisateur.



le programme b_opendoc

Description : permet d'ouvrir un fichier existant ou de créer un nouveau fichier. Les fichiers ouverts sont d'abord sauvegardés dans BiblioBackup, afin qu'il soit possible d'annuler les modifications faites dans ce fichier.

Usage : `b_opendoc doc(string) type_mime`

Portabilité : impossible

si doc ne comporte pas d'extension, utiliser type_mime

note : les type mime sont connus par Windows

le programme b_clic_droit

```
#####
```

```
# fonction b_rightclic
```

```
# b_rightclic operation(string) argument(string) listefichiers ligne1 ligne2(word)...
```

```
# Les noms des fonctions sont dans l'ordre dans biblio.ini [RCLIC]
```

```
# (1) Renommer
```

```
# (2) Supprimer
```

```
# (3) Copier/déplacer
```

```
# (4) Copier sur la disquette
```

```
# (5) Ajouter...
```

```
# (6)
```

```
#
```

```
#####
```

```
{ b_clic_droit operation argument listefichiers ligne1 ligne2...
```

```
operation :      argument :      commentaire :
```

```
- send2disk      USB,dk,CD      ex : envoyer vers la disquette, l'arborescence est reproduite (synchro)
```

```
- mark           on/off         marquer comme important
```

```
- copy           destination    copier un fichier (d'une dossier à un autre)
```

```
- move           destination    déplacer un fichier (d'une dossier à un autre ou backup)
```

```
- delete         destination    utiliser move (copie de sauvegarde)
```

```
- create         nom du fichier créer un fichier
```

listefichier : un fichier texte contenant des chemins de fichiers

ligne1 : un n° de ligne à traiter dans ce fichier

les fichiers supprimés sont en fait déplacés dans biblioBackup et renommés en \$user\$date\$nom_fichier

```
}
```


program verify;

{ Usage : verify

vérifie l'intégrité des fichiers de configuration de biblio :

- le fichier biblio.ini

que se passe-t-il si ?

le fichier biblio.ini n'existe pas : b_configure est lancé par biblio

- les fichiers .box

que se passe-t-il si ?

la première ligne ne contient pas <????>

les chemins ne sont terminés par /

le nombre de ligne n'est pas un multiple de 3 +1

- les fichiers .lst

que se passe-t-il si ?

<fichiers> n'existe pas?

le nombre de lignes à partir de là n'est pas un multiple de 3

- les fichiers log

que se passe-t-il si ?

#mark, #copy, #delete, #create ou #synch n'existe pas

}

Le programme b_indexit

fonctionnalités :

- il scanne les fichiers d'extension StarOffice (sdw), Microsoft Office(doc), Amipro(sam), texte(txt), Rich Text Format (rtf), Writer (wri)
- il en extrait une liste des mots-clés (les plus occurents) et une fiche decriptive (résumé?)
- ces informations sont modifiables manuellement et sont inscrites dans les fichiers .box
- ces informations permettent une recherche de fichiers sur une base thématique.
- les fichiers peuvent être catégorisés : Science, technique, philosophie, religion, etc ... (à définir)

Synchronisation

Biblio en mode serveur exporte son BiblioRoot en « biblio » sous smb,cifs

Biblio en mode client monte « biblio » sous master?