

4D Pack

-  Introduction
-  4D_Pack : ANSI Streams
-  4D_Pack : External Clock
-  4D_Pack : Picture Files
-  4D_Pack : Printing
-  4D_Pack : User interface
-  4D_Pack : Utilities
-  4D_Pack : Windows Help Files
-  Liste alphabétique des commandes

Introduction

 Présentation de 4D_Pack

Présentation de 4D_Pack

Le plug-in 4D_Pack regroupe diverses routines utiles pour le développement de vos applications 4D.

Les commandes de 4D_Pack débutent par le préfixe "AP" afin d'éviter toute confusion avec les commandes propres à 4D. Elles respectent également la convention minuscules/majuscules qui différencie les fonctions (commandes retournant une valeur) des procédures (commandes ne retournant pas de valeur).

Installation

L'installation du plug-in 4D_Pack s'effectue comme pour tout plug-in 4D: il vous suffit de le placer dans le dossier **Plugins** situé près du fichier exécutable de l'application ou dans celui situé près du fichier de structure de la base. Pour plus d'informations, reportez-vous à la section **Ecrire propriétés utilisateur** du manuel Mode Développement.

Note to users of previous versions of 4D_Pack

For reasons related to the evolution of 4D products, several commands found in previous versions have been removed in 4D_Pack v11. Here is the list of deleted commands as well as the recommended replacement solutions:

AP PICT DRAGGER -> Use the integrated drag and drop functions of 4D (MouseDown, MouseX and MouseY **System Variables** in the case of "click" type events)

AP PICT UPDATER and *%AP Pict displayer* -> Use a picture field

AP Read Picture BLOB -> Use **BLOB TO PICTURE**

AP Read Picture File -> Use **READ PICTURE FILE**

AP Save GIF -> Use **CONVERT PICTURE + WRITE PICTURE FILE**

AP SET PICT MODE -> Use **SET FORMAT**

AP Select document -> Use **Select document**

AP Set palette -> Use **Select RGB Color**

AP Sublaunch -> Use **LAUNCH EXTERNAL PROCESS**

AP PrintDefault -> Use *AP BLOB to print settings* + *AP Print settings to BLOB*

AP Text to PrintRec -> Use *AP BLOB to print settings* + *AP Print settings to BLOB*

AP PrintRec to text -> Use *AP BLOB to print settings* + *AP Print settings to BLOB*

AP PrValidate -> Use *AP BLOB to print settings* + *AP Print settings to BLOB*

AP Toolbar installed -> Use **Tool bar height** (returns 0 if toolbar is hidden)

AP SET WEB FILTERS -> Use **SET DATABASE PARAMETER** and **Get database parameter**

AP ShellExecute (Mac OS) --> Use **LAUNCH EXTERNAL PROCESS**

AP Add table and fields -> Use 4D SQL engine

AP Create relation -> No alternative solution currently

4D_Pack : ANSI Streams

 *AP FCLOSE*

 *AP fopen*

 *AP FPRINT*

 *AP fread*

AP FCLOSE

AP FCLOSE (valeurFopen)

Paramètre	Type	Description
valeurFopen	Entier long	→ Référence de canal ou de fichier retournée par la commande AP Fopen

Description

La commande *AP FCLOSE* referme le canal ou le fichier ouvert précédemment par la commande *AP Fopen*.

AP fopen (; mode) -> Résultat

Paramètre	Type	Description
	Chaîne	→ Nom du fichier ou du port qui doit être ouvert
mode	Entier	→ Mode d'ouverture 0 = en écriture seule 1 = en lecture seule
Résultat	Entier long	→ Valeur qui doit être utilisée avec AP FPRINT, ou 0 si le fichier ou le port ne peut pas être ouvert

Description

La commande *AP fopen* appelle la commande C ANSI *fopen*, qui permet d'ouvrir des canaux série ou parallèle (sous Windows), ou encore de créer directement des fichiers.

La valeur retournée par cette commande permet ensuite l'écriture à l'aide la commande *AP FPRINT* dans le canal ou le fichier ouvert. Le fichier ou le canal ouvert peut alors être refermé à l'aide de la commande *AP FCLOSE*.

Note : Sous Mac OS, cette commande permet uniquement de créer des fichiers.

Exemple 1

Écriture de "hello world" dans le port parallèle :

```
port:=AP fopen("LPT1";0)
AP FPRINT(port;"hello world")
AP FCLOSE(port)
```

Exemple 2

Écriture de "hello world" dans le port série :

```
port:=AP fopen("COM1";0)
AP FPRINT(port;"hello world")
AP FCLOSE(port)
```

Exemple 3

Écriture de "hello world" dans le fichier "MonDoc.txt" :

```
port:=AP fopen("C:\MonDoc.txt";0)
AP FPRINT(port;"hello world")
AP FCLOSE(port)
```

Exemple 4

Lecture d'un fichier :

```
port:=AP fopen("C:\MonDoc.txt";1)
MaVar:=AP fread(port)
AP FCLOSE(port)
```

AP FPRINT (valeurFopen ; données)

Paramètre	Type		Description
valeurFopen	Entier long	→	Référence de canal ou de fichier retournée par la commande AP Fopen
données	Texte	→	Données à envoyer

Description

La commande *AP FPRINT* envoie le texte contenu dans le paramètre *données* sur le canal ou le fichier ouvert par la commande *AP Fopen* référencé par *valeurFopen*.

Les données sont envoyées telles quelles, sans conversion. 4D travaillant en interne avec des données en ASCII étendu au format Mac OS, il vous appartient sur PC d'effectuer les conversions éventuelles, en utilisant par exemple la commande **Mac vers Windows**.

AP fread (valeurFopen) -> Résultat

Paramètre	Type		Description
valeurFopen	Entier long	→	Référence de canal ou de fichier retournée par la commande AP Fopen
Résultat	Texte	↩	Texte en provenance du fichier ou du canal

Description

La commande *AP fread* lit le texte en provenance du canal ou du fichier ouvert par la commande *AP Fopen* désigné par le paramètre *valeurFopen*.

Les données sont lues telles quelles, sans conversion. 4D travaillant en interne avec des données en ASCII étendu au format Mac OS, il vous appartient donc sur PC d'effectuer les conversions éventuelles, en utilisant par exemple la commande **Windows vers Mac**.

4D_Pack : External Clock

-  _AP External clock
-  AP SET CLOCK

⚙️ **_AP External clock**

_AP External clock

Ne requiert pas de paramètre

Description

La zone externe *_AP External clock* dessine une horloge analogique en utilisant la ressource PICT 17890 placée dans le plug-in 4D_Pack pour tracer le cadran de l'horloge.

□

Vous pouvez, si vous le souhaitez, dessiner vous-même votre horloge et modifier la PICT 17890 située dans les ressources de 4D_Pack. Cette ressource PICT peut être créée sur un Macintosh, par exemple à l'aide de l'application ResEdit™. L'image peut avoir une taille quelconque, la taille des aiguilles s'adaptera à la taille du cadran que vous avez dessiné. De même, la forme du cadran ne doit pas forcément être contenue dans un carré, mais peut être dans un rectangle quelconque.

Par défaut, l'horloge affiche l'heure du système, mais la commande *AP SET CLOCK* permet de définir un décalage entre l'heure affichée par le cadran et l'heure courante. De cette façon, vous pouvez par exemple afficher une batterie d'horloges correspondant chacune à un fuseau horaire différent.



AP SET CLOCK

AP SET CLOCK (horloge ; heure ; minute)

Paramètre	Type		Description
horloge	Entier long	→	Zone externe _AP External clock
heure	Entier	→	Décalage demandé en heures
minute	Entier	→	Décalage demandé en minutes

Description

La commande *AP SET CLOCK* permet de fixer le décalage de l'horloge affichée par la zone externe *_AP External clock*.
En fixant un décalage par rapport à l'heure du système, vous pouvez afficher l'heure d'un pays situé dans un autre fuseau horaire.

4D_Pack : Picture Files

 *AP Save BMP 8 bits*

AP Save BMP 8 bits

AP Save BMP 8 bits (image ; fichier) -> Résultat

Paramètre	Type		Description
image	Image	→	Image 4D à convertir
fichier	Texte	→	Nom du fichier
Résultat	Entier long	↻	Code d'erreur

Description

Note de compatibilité : Cette commande est désormais obsolète, elle est conservée pour des raisons de compatibilité uniquement et sera supprimée dans les prochaines versions du plug-in. Il est fortement recommandé d'utiliser les commandes du thème "Images" de 4D.

La commande *AP Save BMP 8 bits* permet de convertir et d'enregistrer une image PICT au format BMP en 8 bits non compressés (256 couleurs).

Vous passez dans le paramètre *fichier* le nom et le chemin d'accès du fichier à créer.

Important : Si un fichier de même nom existe déjà à l'emplacement choisi, il est remplacé sans avertissement.

La commande retourne un code d'erreur système en cas de problème, et 0 si tout se passe bien.

4D_Pack : Printing

 AP BLOB to print settings

 AP Print settings to BLOB

AP BLOB to print settings

AP BLOB to print settings (paramImpression ; typeParam) -> Résultat

Paramètre	Type	Description
paramImpression	BLOB	→ BLOB contenant les paramètres d'impression
typeParam	Entier long	→ 0 = configuration et impression, 1 = impression
Résultat	Entier long	↩ 1 = opération réussie, 0 = pas d'imprimante courante, -1 = paramètres incorrects

Description

La commande *AP BLOB to print settings* remplace les paramètres d'impression courants de 4D par les paramètres stockés dans le BLOB *paramImpression*. Ce BLOB doit avoir été généré par la commande *AP Print settings to BLOB*.

Le paramètre *paramImpression* contient deux types de paramètres :

- les paramètres de configuration : papier, orientation, échelle,
- les paramètres d'impression proprement dit : nombre de copies, alimentation, etc.

Vous pouvez choisir de ne charger que les paramètres d'impression. Si vous passez 0 dans *typeParam* ou omettez ce paramètre, tous les paramètres sont utilisés. Si vous passez 1, seuls les paramètres d'impression sont utilisés ; les paramètres de configuration d'impression sont inchangés.

Les nouveaux paramètres d'impression sont appliqués à la totalité de la base et pour toute la session, tant qu'aucune commande telle que **UTILISER PARAMETRES IMPRESSION**, **FIXER OPTION IMPRESSION** ou **IMPRIMER SELECTION** sans le paramètre > ne les modifie.

Les paramètres définis sont notamment utilisés par les commandes **IMPRIMER SELECTION**, **IMPRIMER ETIQUETTES**, **IMPRIMER ENREGISTREMENT**, **Imprimer ligne** et **QR ETAT**, ainsi que par les commandes de menus de 4D, y compris en mode Développement.

Les commandes **IMPRIMER SELECTION**, **IMPRIMER ETIQUETTES**, **IMPRIMER ENREGISTREMENT** doivent impérativement être appelées avec le paramètre > (le cas échéant) afin que le paramétrage défini par *AP BLOB to print settings* soit conservé.

Note : Les paramètres d'impression ne sont pas formatés de la même manière sous Windows et Mac OS. Par conséquent, la compatibilité du BLOB *paramImpression* entre les deux plates-formes n'est pas garantie.

La commande retourne 1 si le BLOB a été correctement chargé, 0 si aucune imprimante courante n'est sélectionnée et -1 si le BLOB est incorrect.

AP Print settings to BLOB

AP Print settings to BLOB (paramImpression) -> Résultat

Paramètre	Type		Description
paramImpression	BLOB	→	BLOB devant recevoir les paramètres d'impression
Résultat	Entier long	↩	Code d'erreur : 1 = opération réussie 0 = pas d'imprimante courante

Description

La commande *AP Print settings to BLOB* stocke les paramètres d'impression courants de 4D dans le BLOB *paramImpression*.
Le paramètre *paramImpression* stocke tous les paramètres utilisés pour l'impression :

- les paramètres de configuration : papier, orientation, échelle,
- les paramètres d'impression proprement dit : nombre de copies, alimentation, etc.

Le BLOB généré ne doit pas être modifié par programmation, il ne peut être exploité que par la commande *AP BLOB to print settings*.

La commande retourne 1 si le BLOB a été correctement généré et 0 si aucune imprimante courante n'est sélectionnée.

4D_Pack : User interface

-  AP FULL SCREEN
-  AP NORMAL SCREEN
-  AP Rect dragger

AP FULL SCREEN

AP FULL SCREEN

Ne requiert pas de paramètre

Description

La commande *AP FULL SCREEN* permet de masquer la barre de menus et la barre de titre de la fenêtre de l'application 4D, et de la passer en plein écran.

Cette commande permet d'exploiter la totalité de l'écran, ce qui est très intéressant par exemple dans le cas d'une application multimédia, d'une borne interactive ou d'un jeu.

Note : Cette commande fonctionne uniquement sous Windows.

AP NORMAL SCREEN

AP NORMAL SCREEN

Ne requiert pas de paramètre

Description

La commande *AP NORMAL SCREEN* permet de faire réapparaître la barre de menus et la barre de titre de la fenêtre de l'application 4D qui avaient été précédemment masquées par la commande *AP FULL SCREEN*.

Note : Cette commande fonctionne uniquement sous Windows.

⚙️ AP Rect dragger

AP Rect dragger (gauche ; haut ; droit ; bas ; x ; y) -> Résultat

Paramètre	Type		Description
gauche	Entier long	→	Coordonnée gauche du rectangle
haut	Entier long	→	Coordonnée supérieure du rectangle
droit	Entier long	→	Coordonnée droite du rectangle
bas	Entier long	→	Coordonnée inférieure du rectangle
x	Entier long	←	Position horizontale d'arrivée la souris
y	Entier long	←	Position verticale d'arrivée de la souris
Résultat	Entier	↪	Numéro du process de destination

Description

Note de compatibilité : A compter de 4D v12, cette commande n'est plus prise en charge sous Mac OS.

La commande *AP Rect dragger* permet de simuler une opération de glisser-déposer ("drag and drop") à partir de 4D. On passe en paramètre les coordonnées de départ du rectangle à faire glisser, et on récupère dans *x* et *y* les coordonnées d'arrivée dans la fenêtre de destination.

Le paramètre retourné donne le numéro du process de destination, ce qui permet de gérer le glisser-déposer interprocess, en appelant par exemple le process de destination par un **APPELER PROCESS**.

Cette commande doit être appelée depuis un bouton invisible.



4D_Pack : Utilities

-  AP Create method
-  AP Does method exist
-  AP Get field infos
-  AP Get table info
-  AP Get templates
-  AP Get tips state
-  AP Modify method
-  AP SET TIPS STATE
-  AP Timestamp to GMT
-  *AP AVAILABLE MEMORY*
-  *AP Get file MD5 digest*
-  *AP GET PARAM*
-  *AP Get picture type*
-  *AP SET PARAM*
-  *AP ShellExecute*

AP Create method (nomMéthode ; tabPropriétés ; codeMéthode ; nomDossier) -> Résultat

Paramètre	Type	Description
nomMéthode	Chaîne	→ Nom de la méthode à créer
tabPropriétés	Tableau entier long	→ Tableau des propriétés (7 valeurs)
codeMéthode	BLOB	→ BLOB contenant le texte de la méthode
nomDossier	Chaîne	→ Dossier de l'Explorateur dans lequel créer la méthode
Résultat	Entier long	↩ Code d'erreur (0=pas d'erreur)

Description

La commande *AP Create method* vous permet d'ajouter une méthode projet dans la structure d'une base de données 4D (bases interprétées uniquement).

Passez le nom de la méthode dans le paramètre *nomMéthode*. Ce nom peut contenir jusqu'à 31 caractères et doit respecter les règles de nommage standard des objets du langage 4D. Si le nom passé comporte plus de 31 caractères, il est tronqué à 31.

Le tableau *tabPropriétés* permet de définir les propriétés de la méthode. Ce tableau doit avoir été déclaré au préalable en Entier long et comporter 7 lignes :

- passez 1 dans *tabPropriétés{1}* si la méthode doit être visible, passez 0 dans le cas contraire.
- passez 1 dans *tabPropriétés{2}* si la méthode doit être disponible via 4DACTION, 4DMETHOD et 4DSCRIPT, passez 0 dans le cas contraire.
- passez 1 dans *tabPropriétés{3}* si la méthode doit être offerte comme Web Service, passez 0 dans le cas contraire.
- si *tabPropriétés{3}* vaut 1, passez 1 dans *tabPropriétés{4}* si la méthode doit être publiée dans le WSDL, passez 0 dans le cas contraire.
- passez 1 dans *tabPropriétés{5}* si la méthode doit être partagée entre les composants et la base hôte, et 0 dans le cas contraire.
- passez 1 dans *tabPropriétés{6}* si la méthode doit être disponible via le SQL, et 0 dans le cas contraire.
- passez 1 dans *tabPropriétés{7}* si la méthode doit disposer de l'attribut "Exécuter sur serveur", et 0 dans le cas contraire.

Passez dans *codeMéthode* un BLOB contenant le texte de la méthode. Si vous utilisez la commande **TEXTE VERS BLOB** pour remplir le BLOB, indiquez le format *Mac Chaîne en C* (ou 0) comme 3e paramètre de cette commande.

Vous pouvez passer dans *nomMéthode* le texte d'une méthode ayant été stockée dans un fichier via la commande de menu **Exporter la méthode...** (dans ce cas, le texte inclut les commandes, constantes, etc., sous forme de références (tokens) et non de mots).

Passez dans *nomDossier* le nom du dossier de l'Explorateur dans lequel la méthode doit être créée. Les dossiers sont visibles dans la page Démarrage de l'Explorateur et permettent de classer les objets suivant des critères personnalisés. Ce paramètre est facultatif ; s'il est omis, la méthode est créée dans le dossier "Méthodes projet par défaut".

Si la méthode est correctement créée, la commande retourne 0. Sinon, elle retourne un code d'erreur. Voici les codes retournés :

- 1 = Un paramètre de la commande est incorrect.
- 2 = Base en mode compilé, impossible de créer une méthode.
- 3 = Le paramètre *nomMéthode* est une chaîne vide.
- 4013 = Le nom de la méthode est invalide.
- 4014 = Impossible de créer la méthode.

Note : Cette commande ne fonctionne pas lorsqu'elle est exécutée depuis une application compilée et fusionnée avec 4D Desktop.

AP Does method exist

AP Does method exist (nomMéthode) -> Résultat

Paramètre	Type		Description
nomMéthode	Chaîne	→	Nom de la méthode à tester
Résultat	Entier	↩	0=la méthode n'existe pas, 1=la méthode existe déjà

Description

La commande *AP Does method exist* permet de savoir si une méthode projet nommée *nomMéthode* existe déjà dans la base de données courante. Cette commande ne tient pas compte des méthodes projet des composants installés dans la base.

Dans le cadre de l'utilisation de la commande *AP Create method*, cette commande a pour but d'éviter l'apparition d'un message d'erreur lorsqu'une méthode de même nom existe déjà.

⚙️ AP Get field infos

AP Get field infos (numTable ; numChamp ; tableLiée ; champLié ; attributs ; choix) -> Résultat

Paramètre	Type		Description
numTable	Entier	→	Numéro de la table
numChamp	Entier	→	Numéro du champ
tableLiée	Entier	←	Related table
champLié	Entier	←	Related field
attributs	Entier long	←	Attributs du champ (mot d'état)
choix	Chaîne	←	Nom d'énumération
Résultat	Entier	↪	Code d'erreur

Description

La commande *AP Get field infos* retourne des informations spécifiques sur le champ 4D dont vous passez les numéros de table et de champ dans les paramètres *table* et *champ*.

Si un lien part du champ, les variables *tableLiée* et *champLié* contiendront les numéros de la table et du champ lié, sinon ces variables prendront la valeur 0.

Chacun des 16 bits de l'entier *attributs* a une signification précise. Vous pouvez lire le contenu de ces bits grâce aux **Opérateurs sur les bits** proposés dans le langage de 4D.

Bit	Vaut 1 si le champ :	Ignoré si le type est
15	est indexé	Texte, Image, Sous-table ou Blob
14	est indexé unique	Texte, Image, Sous-table, Blob, ou si le bit 15 est à 0
13	est obligatoire	Sous-table
12	est énuméré	Image, Sous-table ou Blob
11	est modifiable	Sous-table ou Blob
10	est saisissable	Sous-table ou Blob
8	est invisible	
6	a un lien automatique N vers 1	Texte, Image, Sous-table ou Blob
5	a un lien automatique 1 vers N	Texte, Image, Sous-table ou Blob
2 et 1	a un lien automatique 1 vers N dont l'intégrité référentielle est :	
	00 L'intégrité référentielle n'est pas activée	
	01 Ne modifie pas les enregistrements liés	
	10 Suppression des enregistrements liés	
	11 Interdiction de supprimer des enregistrements liés	
0	a un lien automatique N vers 1 avec mise à jour automatique.	Texte, Image, Sous-table, Blob, ou si le bit 5 n'est pas fixé.

Note : Les bits 3, 4, 7 et 9 sont actuellement inutilisés et réservés.

Si le bit 12 est défini, la variable *choix* contiendra le nom de l'énumération utilisée.

AP Get table info

AP Get table info (numTable ; invisible ; suppression) -> Résultat

Paramètre	Type	Description
numTable	Entier	→ Numéro de table
invisible	Entier	← 1 = table invisible 0 = table visible
suppression	Entier	← 1 = suppression physique des enregistrements 0 = suppression logique des enregistrements
Résultat	Entier	↻ Code d'erreur

Description

La commande *AP Get table info* retourne les attributs de la table 4D dont vous avez passé le numéro dans le paramètre *table*.

La variable *invisible* reçoit 1 si la table a été déclarée invisible en mode Développement, et 0 si la table est visible.

La variable *suppression* reçoit 1 si l'attribut "Définitivement supprimé" (suppression physique) a été sélectionné pour les enregistrements de la table, et 0 sinon (suppression logique).

Rappelons que la suppression physique d'un enregistrement est plus lente que sa suppression logique, mais cet attribut permet d'empêcher sa réapparition en cas de reconstruction de la base par analyse des marqueurs.

AP Get templates (tabNomsModèles)

Paramètre	Type	Description
tabNomsModèles	Tableau chaîne	Noms des modèles

Description

La commande *AP Get templates* remplit le tableau *tabNomsModèles* avec la liste des noms de modèles de formulaires disponibles dans l'application 4D courante.

Les modèles de formulaires sont accessibles dans la boîte de dialogue de l'Assistant de création de formulaires de 4D. Les modèles disponibles par défaut sont par exemple Nostalgie, OSX, OSX (sans libellés), Philadelphia, Sao Paulo, etc. Les formulaires personnalisés construits via les pages avancées de l'Assistant peuvent être ajoutés en tant que modèles.

Cette commande est utile lors de l'utilisation de la commande **AP Add table and fields** afin de récupérer la liste des modèles disponibles.

Note : Cette commande ne fonctionne pas lorsqu'elle est exécutée depuis une application compilée et fusionnée avec 4D Desktop.

Si *AP Get templates* a été exécutée correctement, la commande retourne 0, sinon un code d'erreur est renvoyé.

Exemple

Cet exemple crée une fenêtre listant les modèles disponibles :

```
$ref:=Creer fenetre (<>FGauche;<>FHaut;450;400;8;"Liste des modèles")
TABLEAU ALPHA (255;TabModèles;0)
$error:=AP Get templates(TabModèles)
DIALOGUE ([Table 1];"TabModèles")
FERMER FENETRE ($ref)
```

AP Get tips state

AP Get tips state -> Résultat

Paramètre	Type		Description
Résultat	Entier long		Etat des info-bulles de 4D

Description

La commande *AP Get tips state* retourne 1 si les info-bulles (tips) de 4D sont affichées, sinon elle retourne 0.

AP Modify method

AP Modify method (nomMéthode ; codeMéthode) -> Résultat

Paramètre	Type		Description
nomMéthode	Chaîne	→	Nom de la méthode à modifier
codeMéthode	BLOB	→	BLOB contenant le texte de la méthode
Résultat	Entier long	↻	Code d'erreur (0=pas d'erreur)

Description

La commande *AP Modify method* vous permet de modifier le code source d'une méthode projet dans la structure d'une base de données 4D (bases interprétées uniquement).

Passez le nom d'une méthode projet existante dans le paramètre *nomMéthode* (les noms des méthodes 4D peuvent contenir jusqu'à 31 caractères).

Passez dans *codeMéthode* un BLOB contenant le nouveau texte de la méthode. Si vous utilisez la commande **TEXTE VERS BLOB** pour remplir le BLOB, indiquez le format Mac Chaîne en C (ou 0) comme 3e paramètre de cette commande.

Si la méthode est correctement modifiée, la commande retourne 0. Sinon, elle retourne l'un des codes d'erreur suivants :

- 1 = Un paramètre de la commande est incorrect.
- 2 = Base en mode compilé, impossible de créer une méthode.
- 3 = Le paramètre nomMéthode est une chaîne vide.
- 4013 = Le nom de la méthode est invalide.
- 4015 = Impossible de modifier le code source de la méthode.

Note : Cette commande ne fonctionne pas lorsqu'elle est exécutée depuis une application compilée.

AP SET TIPS STATE

AP SET TIPS STATE (état)

Paramètre	Type	Description
état	Entier →	Affichage des info-bulles 0 = Ne pas afficher les info-bulles 1 = Afficher les info-bulles

Description

La commande *AP SET TIPS STATE* permet d'activer ou de désactiver l'affichage des info-bulles (tips) de 4D.

⚙️ AP Timestamp to GMT

AP Timestamp to GMT (dateLocale ; heureLocale ; dateGMT ; heureGMT) -> Résultat

Paramètre	Type		Description
dateLocale	Date	→	Date locale à convertir
heureLocale	Heure	→	Heure locale à convertir
dateGMT	Date	←	Date GMT retournée par la conversion
heureGMT	Heure	←	Heure GMT retournée par la conversion
Résultat	Chaîne	↪	Heure GMT respectant la norme RFC 822

Description

Note de compatibilité : A compter de la version 12.1 de 4D, la commande *AP Timestamp to GMT* peut être avantageusement remplacée par la commande **Chaîne** de 4D et en utilisant la constante [Date RFC 1123](#). A noter que la commande **Chaîne** permet également de formater les dates et heures selon les principaux standards en vigueur.

La commande *AP Timestamp to GMT* convertit la date et l'heure locales passées dans les paramètres *dateLocale* et *heureLocale* en date et heure GMT, en se basant sur la situation géographique telle qu'elle est définie dans le tableau de bord (date et heure) de la machine. Les nouvelles valeurs sont retournées dans les variables *dateGMT* et *heureGMT*.

En outre, la commande retourne l'heure GMT sous la forme d'une chaîne de caractères qui respecte le format suivant : "jour, JJ MM AAAA HH:MM:SS GMT". Ce format correspond à la norme RFC 822.

Exemple

Nous sommes jeudi 13 novembre 1997 et il est 14 heures 35 (heure de Paris). L'instruction suivante retourne la chaîne "Thu, 13 Nov 1997 13:35:00 GMT". La variable *\$gmtd* prend la valeur 13/11/97 et la variable *\$gmtt* prend la valeur 13:35:00.

```
$$:=AP Timestamp to GMT(Date du jour;Heure courante;$gmtd;$gmtt)
```



AP AVAILABLE MEMORY

AP AVAILABLE MEMORY (mémoireTotale ; mémoirePhysique ; mémoireLibre ; pileDisponible)

Paramètre	Type		Description
mémoireTotale	Entier long	←	Mémoire totale sur la machine
mémoirePhysique	Entier long	←	Mémoire physique sur la machine
mémoireLibre	Entier long	←	Mémoire disponible pour 4D
pileDisponible	Entier long	←	Pile disponible

Description

Note de compatibilité : Cette commande est désormais obsolète, elle est conservée pour des raisons de compatibilité uniquement et sera supprimée dans les prochaines versions du plug-in.

La commande *AP AVAILABLE MEMORY* retourne des informations sur la mémoire installée et la mémoire disponible sur la machine où elle est exécutée. Les valeurs retournées sont exprimées en kilo-octets.

La commande retourne la mémoire installée sur la machine (totale et physique), la mémoire disponible pour 4D et la pile disponible (arrondi au Ko près) du process courant.

AP Get file MD5 digest

AP Get file MD5 digest (cheminFichier ; digest ; fork) -> Résultat

Paramètre	Type		Description
cheminFichier	Texte	→	Chemin complet du fichier
digest	Texte	←	Digest MD5 du fichier
fork	Entier long	→	0=Data fork, 1=Resource fork
Résultat	Entier long	↻	Code d'erreur

Description

Note de compatibilité : A compter de 4D v13, la commande **AP Get file MD5 digest** est obsolète et ne doit plus être utilisée. Elle doit être remplacée par la commande 4D **Generer digest**. **AP Get file MD5 digest** est maintenue pour des raisons de compatibilité uniquement.

AP GET PARAM

AP GET PARAM (option ; valeur)

Paramètre	Type		Description
option	Entier	→	Numéro de l'option à lire
valeur	Entier	←	Valeur de l'option lue

Description

Note de compatibilité : Cette commande est désormais obsolète, elle est conservée pour des raisons de compatibilité uniquement et sera supprimée dans les prochaines versions du plug-in. Il est fortement recommandé d'utiliser la commande [Lire parametre base](#) de 4D.

La commande *AP GET PARAM* permet de lire les options internes de 4D, telles qu'elles peuvent être paramétrées à l'aide de 4D Customizer Plus.

L'intérêt de cette commande est principalement de pouvoir remettre un paramètre à sa valeur d'origine après modification.

Pour plus d'informations sur le paramètre *valeur*, reportez-vous à la description de la commande *AP SET PARAM*.

Exemple

Cette instruction vous permet d'obtenir la valeur courante du paramètre 2 (vitesse de rotation du curseur en forme de roue) :

```
AP GET PARAM(2;$param)
```

⚙️ AP Get picture type

AP Get picture type (imageStockée) -> Résultat

Paramètre	Type	Description
imageStockée	BLOB	Image stockée dans un BLOB
Résultat	Entier long	Type de l'image

Description

Note de compatibilité : Cette commande est désormais obsolète, elle est conservée pour des raisons de compatibilité uniquement et sera supprimée dans les prochaines versions du plug-in. Il est fortement recommandé d'utiliser les commandes du thème "Images" de 4D.

La commande *AP Get picture type* retourne le type de l'image stockée dans le BLOB *imageStockée*.

Les valeurs retournées par la fonction peuvent être les suivantes :

- 1 Type inconnu
- 0 Type PICT
- 1 Image PICT avec compression QuickTime
- 2 Type JPEG
- 3 Type WMF
- 4 Type EMF
- 5 Type BMP
- 6 Type GIF

Exemple

Cet exemple stocke une image dans un BLOB puis affecte la valeur du type à la variable *\$type*.

```
DOCUMENT VERS BLOB($fichierImage;$LeBLOB)
$type:=AP Get picture type($LeBLOB)
```

AP SET PARAM

AP SET PARAM (option ; valeur)

Paramètre	Type		Description
option	Entier	→	Numéro de l'option à modifier
valeur	Entier	→	Nouvelle valeur de l'option à modifier

Description

La commande *AP SET PARAM* permet de modifier les paramètres du programme enregistrés dans la ressource CUST ID=0. Pour cela, vous passez le numéro de l'option à modifier dans le paramètre *option* et sa nouvelle valeur dans le paramètre *valeur*. La ressource CUST ID=0 est une suite de valeurs entières structurée de la façon suivante (le n° de l'entier est suivi de sa signification) :

N°	Valeur défaut	Signification
0	0	0 = il n'est pas possible d'utiliser les accessoires de bureau durant une impression 1 = cela est possible
1	0	0 = il n'est pas possible d'utiliser les accessoires de bureau durant l'affichage du thermomètre (recherches, tris...) 1 = cela est possible Pour la version Windows, la valeur -4 signifie que 4D va utiliser un buffer offscreen permanent, ce qui optimise notablement les affichages au détriment d'une occupation mémoire plus importante.
2	0	Fixe la vitesse de rotation du curseur en forme de roue lors de l'exécution de formules 4D. 0 = le curseur tournant n'apparaît pas Nombre positif ou négatif non nul = vitesse de rotation modulo 32 Valeur moyenne pouvant être passée = 16.
3	(voir ci-contre)	Contient le code ASCII du caractère générant un nouveau sous-enregistrement lors de la saisie d'un enregistrement. Valeur par défaut sous Mac OS = 3 (touche Entrée). Valeur par défaut sous Windows = 58 (touche [:])
4	(voir ci-contre)	Contient la valeur indiquant la touche de modification associée au caractère indiqué par l'entier n°3. Valeur par défaut sous Mac OS = 256 (touche Commande). Valeur par défaut sous Windows = 0 (pas de touche). Valeurs possibles = 256 pour la touche Commande/Ctrl, 512 pour la touche Majuscule, 1024 pour la touche Verrouillage Majuscule, 2048 pour la touche Option/Alt.
5	3 (Entrée)	Contient le code ASCII du caractère générant la validation de la saisie d'un enregistrement.
6	(voir ci-contre)	Contient la valeur indiquant la touche de modification associée au caractère indiqué par l'entier n°5. Valeur par défaut sous Mac OS = 2048 (touche Option). Valeur par défaut sous Windows = 0 (pas de touche)
7	27 (Echap)	Contient le code ASCII du caractère générant l'annulation de la saisie d'un enregistrement.
8	(voir ci-contre)	Contient la valeur indiquant la touche de modification associée au caractère indiqué par l'entier n°7. Valeur par défaut sous Mac OS = 2048 (touche Option). Valeur par défaut sous Windows = 0 (pas de touche)
9	3 (Entrée)	Contient le code ASCII du caractère générant la validation d'une boîte de dialogue.
10	0 (pas de touche)	Contient la valeur indiquant la touche de modification associée au caractère indiqué par l'entier n°9.
11	27 (Echap)	Contient le code ASCII du caractère générant la non-validation d'un dialogue.
12	0 (pas de touche)	Contient la valeur indiquant la touche de modification associée au caractère indiqué par l'entier n°11.
13	1	Indique l'emplacement par défaut de la fenêtre de 4D. Les valeurs possibles sont : 0 = la fenêtre à la taille de l'écran avec sa barre de titre visible en-dessous de la barre des menus. 1 = la fenêtre a la taille de l'écran avec sa barre de titre cachée sous la barre des menus. 2 = la fenêtre est affichée avec les coordonnées indiquées par les entiers 14,15,16, 17 3 = la fenêtre est affichée avec les coordonnées indiquées par les entiers 14,15,16 et 17 mais est centrée par rapport à l'écran.
14	-1	Indique le côté haut de la fenêtre de 4D.
15	-1	Indique le côté gauche de la fenêtre de 4D.
16	-1	Indique le côté bas de la fenêtre de 4D.
17	-1	Indique le côté droit de la fenêtre de 4D. (Ces quatre entiers ne sont utilisés que si l'entier n°13 contient 2 ou 3)
18	0	Spécifie le redimensionnement de la fenêtre de 4D quand on passe du mode Développement au mode Application. 0 = la fenêtre n'est pas redimensionnée 1 = la fenêtre est redimensionnée suivant la valeur de l'entier n°13
19	0	Spécifie le redimensionnement de la fenêtre de 4D après l'exécution d'une méthode 4D appelée depuis un menu. 0 = la fenêtre n'est pas redimensionnée 1 = la fenêtre est redimensionnée suivant la valeur de l'entier n°13

20		Réservé au programme, ne pas utiliser
21		Réservé au programme, ne pas utiliser
22		Réservé au programme, ne pas utiliser
23		Réservé au programme, ne pas utiliser
24		Réservé au programme, ne pas utiliser
25	0	Indique si le Pendant saisie des formules est appelé lorsque l'utilisateur annule la saisie. 0 = le Pendant saisie est appelé. 1 = le Pendant saisie n'est pas appelé.
26	0	Indique si le mode auto-déverrouillage est actif ou non en mode multi-utilisateurs 0 = le mode n'est pas actif 1 = le mode est actif

Exemple

Cette instruction vous permet d'afficher le curseur en forme de roue lors de l'exécution de vos formules :

```
AP SET PARAM(2;16)
```

AP ShellExecute (nomFichier ; mode ; param) -> Résultat

Paramètre	Type	Description
nomFichier	Chaîne	→ Nom du fichier ou du document à exécuter
mode	Entier long	→ Etat de la fenêtre à l'ouverture
param	Texte	→ Paramètres de l'application
Résultat	Entier long	↻ 0 si l'exécution de la commande est correcte, sinon code d'erreur

Description

Note de compatibilité : Cette commande est désormais obsolète, elle est conservée pour des raisons de compatibilité uniquement et sera supprimée dans les prochaines versions du plug-in. Il est fortement recommandé d'utiliser la commande **LANCER PROCESS EXTERNE** de 4D. A noter que depuis la version 11.0 de 4D, *AP ShellExecute* ne fonctionne plus que sous Windows.

Note : Cette commande fonctionne uniquement sous Windows. Sous Mac OS, elle est inopérante.

La commande *AP ShellExecute* permet de lancer une application ou d'ouvrir un document depuis une base 4D.

Passez dans le paramètre *nomFichier* le nom ou le chemin d'accès complet (absolu) du fichier à lancer. Il est possible de passer uniquement un nom de fichier ou un chemin d'accès relatif lorsque l'application ou le document à ouvrir se trouve à côté de la structure de la base.

Vous pouvez indiquer un nom de document, Windows se chargera d'exécuter l'application associée.

Le paramètre *mode* permet de spécifier le mode d'apparition de la fenêtre dans laquelle l'application sera lancée :

- si vous passez 0, la fenêtre apparaîtra normalement ;
- si vous passez 1, la fenêtre sera en plein écran ;
- si vous passez 2, la fenêtre sera réduite et apparaîtra dans la barre des tâches de Windows.

Le paramètre *param* vous permet de passer tout type de paramètre supplémentaire attendu par l'application exécutée. Par exemple, si vous utilisez cette commande pour lancer un navigateur Web, vous pouvez passer dans ce paramètre un URL spécifique.

Exemple 1

Cet exemple ouvre un document de type Word® se trouvant à la racine du disque principal :

```
`Sous Windows
$err:=AP ShellExecute("C:\Test.doc")
```

Exemple 2

Cet exemple lance le navigateur Internet par défaut en plein écran sur le site de 4D :

```
$err:=AP ShellExecute("www.4d.fr";"1")
```

Exemple 3

Cet exemple ouvre le site de 4D en plein écran sur Netscape® quel que soit le navigateur Internet par défaut :

```
$err:=AP ShellExecute("C:\Program Files\Netscape\NETSCAPE.EXE";"1";"www.4d.fr")
```

Exemple 4

Cet exemple ouvre l'Explorateur Windows dans une fenêtre normale et affiche le contenu du dossier Music :

```
$err:=AP ShellExecute("C:\WINDOWS\explorer.exe";0;"C:\Music")
```

4D_Pack : Windows Help Files

-  AP CLOSE HELP
-  AP HELP INDEX
-  AP HELP ON HELP
-  AP HELP ON KEY

AP CLOSE HELP

AP CLOSE HELP

Ne requiert pas de paramètre

Description

Note de compatibilité : Cette commande n'est pas compatible avec Windows Vista. Son usage est désormais déconseillé.

La commande *AP CLOSE HELP* referme l'application d'aide en ligne standard de Windows.

Note : Cette commande fonctionne uniquement sous Windows.



AP HELP INDEX (aide)

Paramètre	Type	Description
aide	Chaîne	Nom de fichier d'aide

Description

Note de compatibilité : Cette commande n'est pas compatible avec Windows Vista. Son usage est désormais déconseillé.

La commande *AP HELP INDEX* affiche la liste des mots-clés disponibles pour le fichier d'aide en ligne Windows passé en paramètre.

Note : Cette commande fonctionne uniquement sous Windows.

AP HELP ON HELP

AP HELP ON HELP

Ne requiert pas de paramètre

Description

Note de compatibilité : Cette commande n'est pas compatible avec Windows Vista. Son usage est désormais déconseillé.

La commande *AP HELP ON HELP* affiche l'aide en ligne propre à l'application d'aide en ligne standard de Windows.

Note : Cette commande fonctionne uniquement sous Windows.

AP HELP ON KEY (aide ; motClé)

Paramètre	Type		Description
aide	Chaîne	→	Nom du fichier d'aide
motClé	Chaîne	→	Mot-clé pour lequel afficher l'aide

Description

Note de compatibilité : Cette commande n'est pas compatible avec Windows Vista. Son usage est désormais déconseillé.

La commande *AP HELP ON KEY* ouvre le fichier d'aide passé dans le paramètre *aide* avec l'application d'aide en ligne standard de Windows, et affiche la page qui correspond au mot-clé.

Note : Cette commande fonctionne uniquement sous Windows.

