9

L'environnement graphique de Prolog IV

CE CHAPITRE décrit brièvement l'environnement graphique de Prolog IV. Outre la description des différents menus, fenêtres et dialogues, on y trouvera la liste des primitives graphiques accessibles depuis Prolog IV.

9.1 Lancement de Prolog IV

L'environnement graphique de Prolog IV est construit autour de l'outil TCL/TK, appartenant au domaine public. TCL est un langage de programmation et TK sa librairie graphique, originellement conçu pour X11 [1]. Il faut donc que X11, Motif ou quelque serveur compatible X soit en fonctionnement sur votre terminal avant de pouvoir lancer l'environnement graphique. Si c'est le cas, on peut exécuter la commande `prolog4e` dans une fenêtre « shell », comme `xterm`. L'environnement lancé ouvre alors un panneau avec boutons, et une fenêtre console.

Du texte doit alors s'afficher en haut de cette fenêtre, ressemblant à ceci, aux dates et numéro de version près :

```
[Connection OK]
Prolog IV v1.0 (222), Juillet 1996 (C)PrologIA 1995,1996
[ Fri Jun 14 10:45:43 MET DST 1996 ]

>>
```

Si aucun texte ne s'affiche dans cette fenêtre, c'est que :

- la fin de l'installation (configuration) n'a pas été effectuée complètement ;
- il n'y a pas assez de mémoire pour faire tourner Prolog IV.
    - Peut-être plusieurs Prolog IV tournent-ils sur cette machine ?
    - Peut-être les tailles de piles sont-elles trop fortes pour la configuration mémoire de votre machine ? Il faut alors les réduire, dans le dialogue de préférences, sous le thème « Prolog ».

1. Il est maintenant disponible sur les différents Windows xx de la société Microsoft, ainsi que sur Macintosh, d'Apple Corp.

9.2 Configuration initiale

Ceci est à effectuer une fois au moins, généralement lors de l'installation.

Il faut configurer les paramètres en rapport avec Prolog IV, et ce dans le dialogue des préférences, sous le thème « Prolog ». Le minimum requis est de mettre dans le champ « Exec Name: » le chemin de l'exécutable Prolog IV, et de cocher les boutons «Uses Tcl» et «Auto Launch». Les substitutions ~ du shell Unix sont acceptées au début du chemin.

Pour que cette configuration soit conservée pour d'autres sessions, il faut l'enregistrer en activant l'item «Save Preferences» du menu «Info» du panneau principal.

L'exécutable Prolog IV peut alors être lancé en activant l'item « Run » du menu «Process» de la fenêtre intitulée «Console Prolog IV».

On peut aussi quitter l'environnement graphique de Prolog IV et le relancer.

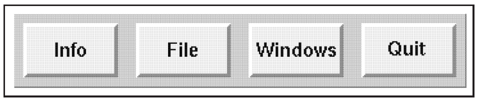
9.3 Le panneau principal

L'environnement graphique se présente sous la forme d'un panneau principal, de dialogues et de fenêtres d'édition appelés «éditeurs».

Le panneau principal regroupe les diverses parties de l'environnement graphique, soit l'accès aux éditeurs, à diverses informations et à Prolog IV, fonctionnant dans une fenêtre appelée «Console Prolog IV».

D'un certain point de vue, cet environnement n'est pas lié à Prolog IV, et il est théoriquement possible de lancer tout autre programme que Prolog IV utilisant les entrées/sorties standard d'Unix. Cependant, un bon nombre de fonctions sont largement orientées vers une utilisation Prolog.

Voici la description des différents éléments du panneau principal :



9.3.1 Le menu «Info»

Le menu info regroupe divers types d'information comme l'accès à un «Lisez-Moi», aux préférences, à un dialogue pour la rédaction de rapports de bugs, etc.

A propos de Prolog IV ... : affiche la version courante de Prolog IV.

Preferences ... : accès au dialogue de Préférences.

Save Preferences : sauve les préférences dans le répertoire HOME de l'utilisateur actuel.

Credits ... : quelques remerciements concernant certaines parties du logiciel.

Bug Report ... : crée un dialogue «normalisé» pour envoyer des descriptifs de problèmes rencontrés ou des suggestions concernant le logiciel.

Read Me ... : affiche une fenêtre contenant des informations sur l'environnement graphique (dont certaines de dernière minute). C'est un «Lisez-Moi».

9.3.2 Le menu «File»

Le menu fichier donne accès à quelques fonctionnalités générales concernant les éditeurs de texte.

New : crée une nouvelle fenêtre d'édition, vide.

Open ... : propose un dialogue d'ouverture de fichier, et ouvre celui choisi dans une nouvelle fenêtre.

Save All : sauve tous les éditeurs modifiés. N'interroge le programmeur que lorsqu'un éditeur n'avait jamais été précédemment sauvé.

Close Unedited : ferme les éditeurs non-modifiés (les détruit).

9.3.3 Le menu «Windows»

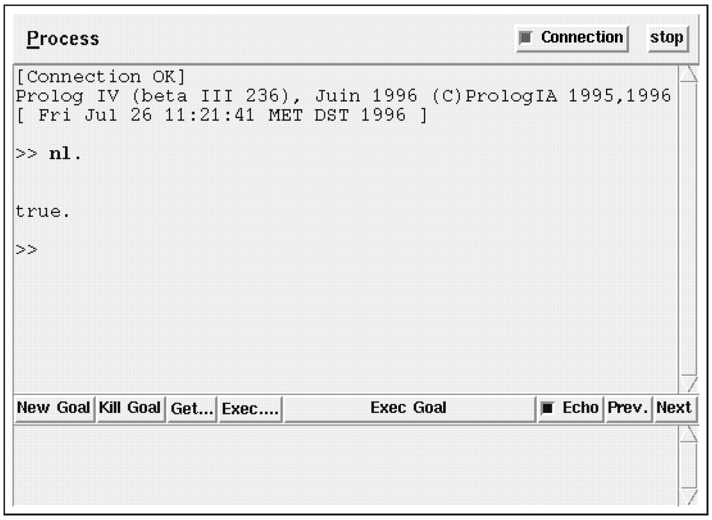
Le menu fenêtres ne montre à ce jour que les éditeurs ouverts. La console n'y figure donc pas. Cliquer dans un des items du menu fait venir la fenêtre de ce nom au premier plan (éventuellement en la faisant passer de l'état d'icône à l'état de fenêtre ouverte).

9.3.4 Le bouton «Quit»

Ce bouton permet de quitter l'environnement de développement Prolog IV. Le cas échéant, un dialogue demande de sauver les modifications effectuées dans les éditeurs.

9.4 La console Prolog IV

La console est la fenêtre qui permet au programmeur d'interagir avec Prolog IV. En plus de la barre de menu, elle contient un panneau de texte éditable qui affiche les entrées et les sorties de Prolog IV, ainsi qu'un panneau de texte qu'on appelera «Buts» permettant de conserver des buts qu'on utilise fréquemment.



La barre de menu se compose du menu et des boutons suivants :

9.4.1 Le menu « Process »

Run : lance l'exécutable désigné dans le champ « Exec. Name: » situé dans les préférences « Prolog », en lui concaténant les options du champ « Options: », puis la chaîne composée de deux tirets « −− », puis les options du champ « User-Options ». « Run » termine l'exécution d'un éventuel processus lancé préalablement dans la console, sans demander confirmation (en utilisant « Kill »).

Kill : termine brutalement l'exécution du processus lancé dans la console. Pour les curieux, c'est un signal SIGTERM qui est généré sous UNIX.

Preferences ... : donne l'accès directement aux préférences en rapport avec Prolog IV (ou plus généralement l'exécutable à lancer).

9.4.2 Le bouton « Stop »

Ce bouton envoie une interruption utilisateur au processus tournant dans la console. Dans le cas de Prolog IV, un but en cours d'exécution sera interrompu, avec un retour au prompt, à moins que le programme Prolog IV n'intercepte cette interruption. Pour les curieux, c'est un signal SIGINT qui est généré sous UNIX.

9.4.3 Le bouton « Connection »

Il est utile de savoir que s'il est coché, c'est que Prolog IV tourne effectivement dans la console, et que ses entrées/sorties standard y seront saisies ou affichées.

Un autre bouton « Connection » est présent dans la fenêtre du débogueur.

Note : la sortie « stderr » d'un processus ne peut être montrée.

9.4.4 Le panneau des buts

Le panneau « Buts » comporte une partie texte (avec ascenseur) et quelques boutons et menus. Ce panneau permet d'éditer des buts, de les lancer et de se déplacer parmi eux. L'ensemble des buts de ce panneau est organisé sous la forme d'une file. On peut insérer et supprimer des buts.

New Goal : crée un nouveau but. Il s'insère après celui présenté.

Kill Goal : détruit le but présenté dans la zone texte. Le but suivant (s'il existe) est alors visualisé à sa place (sinon c'est le précédent).

Get ... : visualise dans (pour une éventuelle édition) la zone texte le but choisi dans ce menu.

Exec ... : lance (sans le mettre dans la zone texte) le but choisi dans ce menu.

Exec Goal : lance le(s) buts contenu(s) dans la zone texte. En fait, ceci envoie tout texte contenu dans cette zone à Prolog, sans la moindre vérification. On peut dont y mettre aussi bien des requêtes que des entrées.

Echo : quand ce bouton est coché, les buts lancés sont affiches dans la console.

Prev. : visualise le but suivant dans la liste des buts. Emet un bip s'il n'existe pas.

Next : visualise le but précédent dans la liste des buts. Emet un bip s'il n'existe pas.

Des raccourcis-clavier existent pour certains boutons (attention aux majuscules et minuscules) :

- `ENTER` (du pavé numérique) pour le bouton « Exec Goal » qui envoie donc le but courant dans la console,
- `CTRL-p` pour le bouton « Prev. »,
- `CTRL-n` pour le bouton « Next »,
- `CTRL-N` pour le bouton « New Goal »,
- `CTRL-D` pour l'action «Dupliquer le but actuellement présenté» (effectue donc un « New Goal »),
- `CTRL-K` pour le bouton « Kill Goal ».

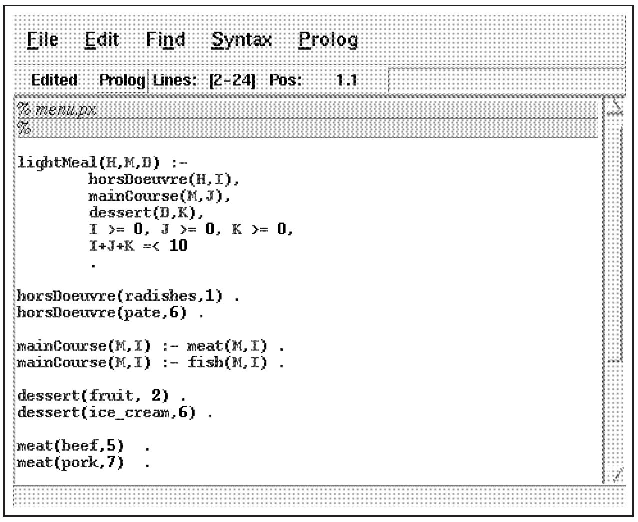
9.5 La Console Tcl/Tk

Les utilisateurs ayant une certaine connaissance du langage TCL peuvent l'utiliser.

La console Tcl/Tk permet d'exécuter des commandes TCL. Il suffit, après avoir tapé la commande, de la sélectionner et de presser le bouton « Eval Selection ». On peut aussi, si la commande ne tient que sur une seule ligne (qui est éventuellement lovée sur plusieurs lignes physiques), faire « `Enter` » (du pavé numérique), le point d'insertion étant dans cette ligne. A l'heure actuelle, les versions utilisées de cet outil sont les suivantes : Tcl 7.3 et Tk 3.6, ainsi que TclX 7.3 .

9.6 Les Editeurs

Chaque fenêtre dispose de sa propre barre de menu, ainsi qu'une ligne comportant quelques informations, placée sous la barre. Le titre de la fenêtre porte le nom du fichier (Untitled-n s'il n'a pas encore été sauvé) et son chemin.



La ligne de statut contient les éléments suivants, de la gauche vers la droite :

– Un statut du fichier édité (rien ou «Edited» si le fichier n'a pas été sauvé depuis la dernière modification).

– Un menu indiquant le langage de programmation utilisé dans d'éditeur (à positionner soi-même). Ceci n'a d'importance que pour la coloration syntaxique dynamique.

– L'intervalle de lignes visibles dans la fenêtre.

– La position actuelle du curseur d'insertion, sous la forme $ligne.colonne$ (les lignes commencent à 1, les colonnes à 0).

– Une éventuelle action en cours d'exécution.

La barre de menu se compose des menus «File», «Edit», «Find», «Syntax», «Prolog» décrits ci-après.

9.6.1 Le menu «File»

New Window... : crée un nouvel éditeur.

Clear All : efface le contenu de l'éditeur. Demande confirmation s'il y a lieu.

Load File ... : présente un dialogue d'ouverture de fichier à charger dans la fenêtre courante en remplacement du fichier déjà présent. Demande confirmation s'il y a lieu.

Save : sauve le contenu de la fenêtre dans un fichier. La première fois, un nom sera demandé.

Save As ... : sauve le contenu de la fenêtre dans un fichier dont le nom sera demandé. Renomme la fenêtre le cas échéant.

Save To ... : sauve le contenu de la fenêtre dans un autre fichier, dont le nom sera demandé.

Close : ferme la fenêtre. Demande confirmation s'il y a lieu.

9.6.2 Le menu «Edit»

Le menu «Edit» ressemble beaucoup à ce qui se fait sur Macintosh ou NeXT.

? Undo : n'est pas encore implanté.

Cut : copie la sélection de la fenêtre (si elle existe) dans un «presse-papier» puis détruit cette sélection.

Copy : copie la sélection de la fenêtre (si elle existe) dans un «presse-papier»

Paste : colle le contenu du «presse-papier» au point d'insertion. Ceci ne remplace pas la sélection.

Paste Selection : colle le texte d'une sélection (de quelque fenêtre X11 qu'elle vienne) au point d'insertion. Ceci ne remplace pas la sélection.

Clear : détruit cette sélection.

Shift Left : décalage à gauche (d'un tab) des lignes de la sélection.

Shift Right : décalage à droite (d'un tab) des lignes de la sélection.

Misc. : accès à un sous-menu permettant d'établir ou non l'auto-indentation des lignes lors de la frappe d'un retour-chariot ou d'établir la coloration syntaxique dynamique.

Select All : sélectionne tout le contenu de la fenêtre.

Font list : accès à un sous-menu permettant de choisir la police de caractère à utiliser dans la fenêtre.

9.6.3 Le menu «Find»

Le menu « Find » permet d'effectuer des recherches et des remplacements, en tenant compte ou en ignorant la casse[2], en recherchant des expressions régulières. Le dialogue «Find/Replace» peut être activé par l'item «Find ...» du menu ; il est décrit plus loin.

Find ... : appelle le dialogue de Recherche/Remplacement.

Find Next : recherche et sélectionne la prochaine occurrence.

Find <Selection> : recherche et sélectionne la prochaine occurrence de la chaîne sélectionnée.

Replace : remplace la sélection par la chaîne de substitution.

Replace & Find : remplace la sélection par la chaîne de substitution puis cherche la prochaine occurrence.

Replace All : effectue le remplacement dans tout le fichier.

Find All : colorie toutes les occurrences de la chaîne à chercher (ce n'est pas une sélection, juste une coloration sans incidence sur l'édition !).

Next Found : fait défiler le texte pour montrer la prochaine occurrence colorée dans la fenêtre.

Unhighlight All Found : enlève la coloration des textes trouvés par des recherches antérieures (voir «Find all»).

Selection – > Find-field : place le texte de la sélection courante dans le champ «Find» du dialogue de Recherche/Remplacement.

Selection – > Replace-field : place le texte de la sélection courante dans le champ «Replace» du dialogue de Recherche/Remplacement.

9.6.4 Le menu «Syntax»

Check Syntax : vérifie la syntaxe du contenu de la fenêtre. Présente les lignes erronées ou comportant un avertissement dans un style (couleur) spécial (ce n'est pas une sélection, juste une coloration). On peut se servir ensuite de « Goto Next Error » pour les visualiser. Passer la souris sur des caractères de ce style fait apparaître les messages d'erreur ou d'avertissements en rapport avec cet endroit.

2. l'attribut «majuscule» ou «minuscule» d'une lettre.

$\boxed{\text{Check Syntax (Selection)}}$: même chose que précédemment, mais ne marche que pour le contenu de la sélection.

$\boxed{\text{Goto Next Error}}$: rend visible à l'écran une éventuelle prochaine ligne colorée par «Check Syntax». Arrivé sur la dernière erreur, fait un bip et repasse à la première erreur du fichier.

$\boxed{\text{Clear All Error Marks}}$: ôte de tout le fichier les styles spéciaux «erreur» et «avertissement» des portions de texte soupconnées comme étant erronées ou ayant donné lieu à des avertissements.

$\boxed{\text{Clear Warning Marks}}$: ôte de tout le fichier le style spécial « avertissement » des portions de texte ayant donné lieu à des avertissements.

$\boxed{\text{Syntax Coloring}}$: colorie tout le fichier (entités syntaxiques et règles). *C'est pour l'instant encore expérimental.* On ne peut pas l'enlever simplement après coup.

9.6.5 Le menu «Prolog»

ce menu permet l'accès aux commandes principales d'ajout de règles à l'exécutable Prolog IV, si celui-ci est en fonctionnement. Le compilateur Prolog IV est utilisé. Le mode (re)consult est disponible, au travers d'un sous-menu. Par défaut, toutes les opérations qui suivent font d'abord appel à la vérification syntaxique.

$\boxed{\text{Check Syntax before}}$: si cet item est coché, l'activation des items décrits ci-après entrainera préalablement la vérification syntaxique de tout le fichier (ou toute la sélection). S'il y a des erreurs, un mini-rapport est présenté dans un dialogue et l'action en cours (comme compile) est automatiquement annulée. Il n'est pas tenu compte de la présence d'avertissements (warning).

$\boxed{\text{Debug}}$: si cet item est coché, les compilations ultérieures de texte Prolog dans tous les éditeurs se feront en mode «debug».

$\boxed{\text{Compile Window}}$: compile le contenu de la fenêtre (pas du fichier).

$\boxed{\text{Compile Selection}}$: compile le contenu de la sélection de la fenêtre.

$\boxed{\text{Recompile Window}}$: recompile le contenu de la fenêtre (pas du fichier).

$\boxed{\text{Recompile Selection}}$: recompile le contenu de la sélection de la fenêtre.

$\boxed{\text{Consulting}}$: permet d'accéder aux commandes de consultation de fenêtre.

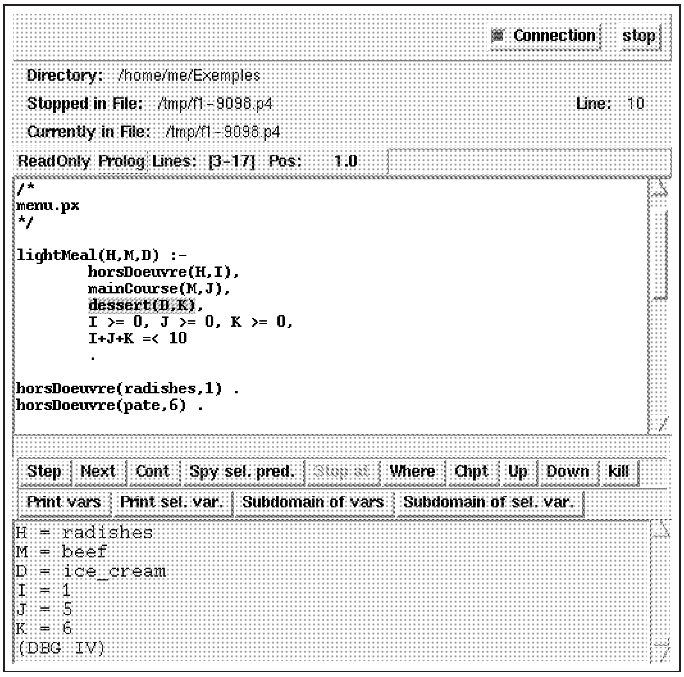
9.7 Le Débogueur

Ce dialogue est en fait une interface avec le débogueur Prolog IV décrit dans le chapitre «Environnement». Elle offre en plus la visualisation du source de ce qui a été compilé (en mode `debug`) pendant l'exécution.

9.7.1 Constitution

Des boutons du panneau central facilitent l'usage des commandes du débogueur, en évitant à l'utilisateur de s'en souvenir et d'avoir à les taper. Les commandes les plus usuelles ont été associées à des boutons.

Cliquer sur un de ces boutons équivaut en général à taper le texte correspondant à la commande dans l'entrée courante. Il faut donc que le débogueur Prolog IV soit en attente de lecture d'une commande.



Cette fenêtre comporte trois parties principales, qui sont :

– une zone de texte dans laquelle un source prolog peut être affiché, par le débogueur.

– un panneau de boutons, effectuant des commandes du débogueur.

– un zone de texte pouvant contenir les entrées/sorties du débogueur ; son invite s'y affiche.

Un bouton à cocher, nommé Connection indique si les sorties (que ce soient celles du débogueur ou les réponses aux requêtes) se font dans cette fenêtre plutôt que dans la console Prolog IV.

9.7.2 Usage

Pour que le débogueur source soit utilisable, il faut avoir compilé le contenu des fenêtres en mode `debug`, en positionnant l'item «Debug» du menu «Prolog » de l'un des éditeurs. On actionne ensuite l'un des items de compilation (pas de consultation !) de ce même menu (« compile xxx » ou « recompile xxx »).

Dans la console Prolog IV, il faut avoir exécuté la primitive `debug.` pour se mettre dans le mode du même nom, avant de taper une requête. Pendant l'exécution de la requête, la fenêtre du débogueur vient se placer au premier plan dès que des portions compilées en mode `debug` viennent à être exécutées. Il faut parfois effectuer plusieurs fois l'action `step` (ou le bouton de même nom) avant que le source ne soit affiché, certains buts (comme les primitives prédéfinies) n'étant pas montrables au niveau source.

9.8 Dialogues

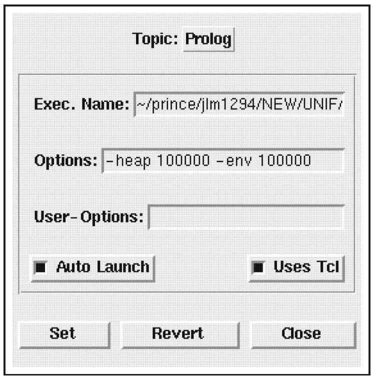
Sans aller jusqu'à décrire tous les dialogues présents dans l'environnement graphique de Prolog IV, présentons les principaux dans les sections suivantes.

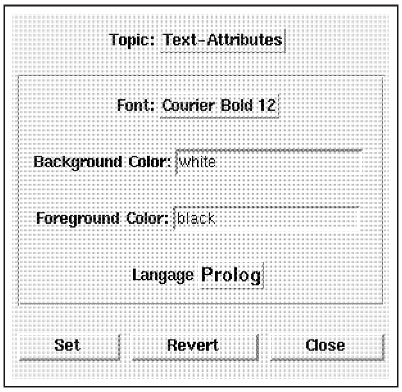
9.8.1 Préférences

Ce dialogue permet, par le biais de deux sous-panneaux qui repésentent chacun un thème, de :

– positionner le paramétrage de l'exécutable Prolog IV lancé dans la fenêtre Console ;

– choisir quelques attributs généraux des éditeurs de textes.





Dans tous les cas, les boutons Set, Revert et Close sont présents. Voici leurs significations :

Set : valide les valeurs des options positionnées.

Revert : remet les options à leurs dernières valeurs enregistrées par Set.

Close : ferme la fenêtre ; les modifications non validées par Set sont perdues.

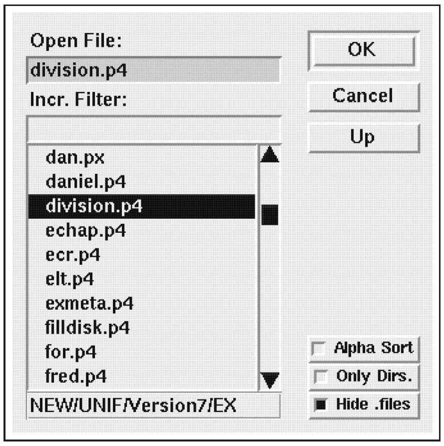
Changer de thème perd également les modifications non validées par Set.

Pour enregistrer ces divers paramètres d'une session à l'autre, il faut actionner l'item « Save Preferences » du menu « Info » du panneau principal.

9.8.2 Boîte d'ouverture/enregistrement de fichier

Ce dialogue permet de choisir un fichier à charger dans un éditeur. Ce même dialogue permet de donner un nom à un fichier que l'on veut enregistrer sur disque.

Voici les différentes parties de ce dialogue.



Les champs éditables :

Open File: : ou

Load File: : ou

Save File: : selon la nature de l'opération, contient le nom de fichier à charger ou à enregistrer.

Incr. Filter: : permet de composer incrémentalement un nom de fichier. Au fur et à mesure de la frappe, les noms de fichiers conformes au filtre courant sont affichés dans la liste des fichiers.

Une liste déroulable de fichiers du répertoire courant (et conformes au filtrage courant) est affichée. On peut bien entendu double-cliquer sur un nom de fichier pour confirmer son choix. Dans le cas d'un double-clic sur le nom d'un répertoire (une marque / est placée devant un nom de répertoire), on est placé dans celui-ci et la liste est mise à jour en fonction de son contenu. Un double-clic équivaut à faire une sélection et cliquer le bouton OK.

Sous cette liste se trouve dans une étiquette déroulable le chemin complet du répertoire qu'on visite. Cette étiquette n'est pas éditable. On peut le faire défiler avec la souris.

Les boutons :

OK : valide le choix effectué. Dans le cas d'un répertoire, s'y positionne.

Cancel : annule l'opération en cours et fait disparaître le dialogue.

Up : passe au répertoire ancêtre.

Les boutons à cocher, une fois qu'ils le sont, modifient le comportement de la boîte de dialogue de la façon suivante :

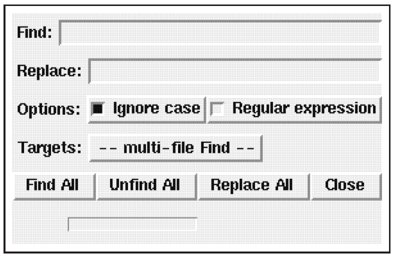
Alpha Sort. : trie les noms alphabétiquement, en ne tenant pas compte des majuscules/minuscules.

Only Dirs. : ne montre que les répertoires.

Hide .files : ne montre pas les fichiers dont le nom commence par le caractère « . » (point) .

9.8.3 Recherche/Remplacement

Ce dialogue permet d'effectuer des recherches et des remplacements de texte dans un ou plusieurs éditeurs de l'environnement.



Les champs éditables :

Find: : contient le texte à rechercher.

Replace: : contient le texte de remplacement.

Des boutons à cocher indiquent le type de recherche :

Ignore Case: : ne tient pas compte des majuscules/minuscules pendant la recherche.

Regular Expression: : une expression régulière sera interprétée dans le champ Find:.

Un menu Targets: permet de choisir les cibles parmi les éditeurs actifs pour les recherches et remplacements. Leurs noms sont dans ce menu. Un ou plusieurs peuvent être sélectionnés.

Les boutons d'actions :

Find All : recherche la chaîne du champ Find: dans toutes les cibles, en colorant les occurrences trouvées.

Unfind All : recherche la chaîne du champ Find: dans toutes les cibles, en décolorant les occurrences trouvées le cas échéant.

Replace All : remplace les occurrences de la chaîne du champ Find: par la chaîne du champ Replace: dans toutes les cibles.

Close : ferme la fenêtre.

Une jauge ainsi que des compteurs d'occurrences (un pour la cible en cours, l'autre pour le cumul en cours) sont présents en bas du dialogue.

9.9 Clavier et souris

9.9.1 La souris

Les deux boutons gauche et droit (gauche et centre si la souris a trois boutons) de la souris sont utilisés sous TCL.

– Le gauche sert toujours aux actions principales.

– Le droit dépend du contexte.

Par exemple, cliquer le bouton droit sur un nom de menu permet de détacher celui-ci.

Dans un texte, maintenir enfoncé le bouton droit en déplaçant la souris permet de se déplacer rapidement. Ceci marche aussi dans un champ éditable.

Sélectionner du texte se fait à l'aide du bouton gauche, comme sur le macintosh (ou le NeXT), avec Shift-clic pour étendre une sélection jusqu'au point cliqué.

Le bouton droit d'une souris à trois boutons n'est a priori pas utilisé.

9.9.2 Touches

Voici quelques informations sur l'utilisation du clavier dans certaines conditions.

Raccourcis-menu

Les raccourcis-menu permettent de naviguer dans les menus en ne se servant que du clavier. Ils se repèrent immédiatement par le soulignement d'un caractère du menu ou de l'un de ses items.

Pour accéder et faire apparaître un menu :

- Taper `MOD-`*lettre* où *lettre* est soulignée dans la barre de menu active.
- Ensuite quand le menu est déroulé :
  - utiliser les flèches pour se déplacer parmi les items, puis taper la touche `Return`.
  - ou bien une lettre pour activer l'item contenant la lettre soulignée.

La touche indiquée par `MOD` est un modifieur qui dépend de votre plate-forme ; c'est la plupart du temps la touche `Alt`. Sur NeXT, c'est la touche «`Command`». Sur SUN c'est la touche «`Alt Graph`».

Raccourcis-clavier

Les raccourcis-clavier permettent d'effectuer une action, sans passer par un menu pour l'activer. Ils sont souvent indiqués à droite de l'item de menu ayant la fonction équivalente.

Effacer la sélection

`Ctrl-d` efface la sélection, de l'objet éditeur ou d'un champ éditable.

Ouverture/Fermeture de menu

La touche `F10` ouvre le premier menu de la fenêtre.

La touche `ESC` ferme un menu déroulé.

9.10 Informations diverses

- Le sommaire du «Read-Me» est un hyper-texte.

– L'analyseur syntaxique effectuant l'opération « Check Syntax » tient compte des directives `op/3` qui figurent dans le fichier. Toutefois, son état interne, en ce qui concerne les opérateurs, reste sans rapport avec l'état interne de Prolog IV si ce dernier a été lancé.

– Les avertissements sont montrés en même temps que les erreurs dans le mini-rapport de « Check Syntax ».

– Passer sur une ligne coloriée comme «erronée» (dans un éditeur) affiche les erreurs ou avertissements en rapport avec cette ligne.

– Il y a une procédure automatique de sauvegarde des éditeurs au contenu non sauvé. Pour chaque éditeur, un fichier «backup» est créé dans `/tmp` et son nom est de la forme `b.ed`N`.frEdit-`PID avec N un entier et PID le numéro de processus de l'environnement graphique auquel appartient l'éditeur. Il n'y a backup que des éditeurs ayant le statut « modifié et non sauvé » (Edited), et ce toutes les cinq minutes. Ce fichier backup est détruit à la fermeture de la fenêtre, et en fin de session.

– Coller quelque chose quand le curseur est dans la sélection courante revient à coller après cette sélection. L'exception est quand le curseur se trouve juste au début de la sélection, auquel cas on colle à cet endroit. Tout ceci dans le but de permettre des séquences de «coller».

– Les fichiers temporaires (créés dans /tmp) sont effacés en fin de session.

9.11 Primitives graphiques

9.11.1 Préalables

Les primitives graphiques fournies permettent de créer des dessins, sous la forme de lignes, rectangles et textes. Ce sont des objets vectoriels, dans le sens où ils ont une certaine indépendance et peuvent être détruits sélectivement. Un programmeur connaissant TCL peut enrichir de façon importante le (petit) jeu de primitives fournies, en modifiant les coordonnées et attributs de ces objets.

Système de coordonnées

On notera les coordonnées par des nombres Prolog IV, qu'ils soient entiers ou flottants. Des rationnels peuvent être également utilisés, quand les valeurs du numérateur et du dénominateur sont «petites»[3].

Le système de coordonnées est défini comme suit, dans chaque panneau graphique :

– L'axe des x est horizontal et croît de gauche à droite.

– L'axe des y est vertical et croît de haut en bas.

– L'origine des axes est dans le coin en haut à gauche du panneau graphique.

3. c.à.d. qu'elles sont manipulables en C.

9.11.2 Primitives

gr_init doit être lancée une fois pour initialiser le graphisme et donc pour permettre l'usage des autres primitives graphiques. Cette primitive crée également la fenêtre graphique, constituée d'un panneau graphique et d'ascenseurs.

gr_moveto(X,Y) déplace le crayon au point de coordonnées (X,Y), qui devient le point courant.

gr_lineto(X,Y) tire une ligne du point courant au point (X,Y).

gr_line(DX,DY) tire une ligne du point courant (X_0,Y_0) au point $X_0 + DX$, $Y_0 + DY$. On transmet donc à cette primitive des variations de coordonnées.

gr_rect(V, [X,Y,X2,Y2]), gr_rect(V, X,Y,X2,Y2) dessinent un rectangle :

- V est l'action (`frame` ou `paint`).
- X,Y sont les coordonnées du coin en haut à gauche.
- $X2$,$Y2$ les coordonnées du coin en bas à droite.

gr_drawstring_at(S, X,Y) pose un objet chaîne contenant S au point (X,Y).

gr_drawstring_at(S, X,Y, X2,Y2) Pose un objet chaîne contenant S dans le rectangle de coins (X,Y) et ($X2$,$Y2$).

gr_arrow(X,Y, X2,Y2) dessine une flèche du point (X,Y) au point destination ($X2$,$Y2$).

gr_setxfont(FX11) positionne la fonte $FX11$ (en employant la dénomination des fontes X11) pour les textes qui seront créés ultérieurement.

gr_setfont(N), gr_setfont(N, T), gr_setfont(N, T, G) positionnent la fonte des textes qui seront créés ultérieurement :

- N est le nom (helvetica, times, courier, etc)
- T est la taille (12, 14, 18, 24, etc)
- G est la graisse (bold, medium, etc)

D'autres fontes peuvent être disponibles sur une plate-forme donnée.

gr_fgcolor3(R, G, B) positionne la couleur des objets ultérieurement créés à la valeur #RGB, une chaîne représentant une couleur et formée des composantes rouge, verte et bleue données comme arguments. C'est une convention X11 ; par exemple, #ff0000 désigne le rouge, #ffffff est le blanc, et #000000 est le noir.

gr_fgcolor(Coul) positionne la couleur des objets ultérieurement créés à $Coul$, qui doit être un nom de couleur X11 (comme aquamarine, slategray, etc).

gr_settag(Tag) positionne l'étiquette Tag (un atome), pour les prochains objets à créer. La valeur spéciale `{}` l'enlève.

gr_erase détruit tous les objets de la fenêtre graphique courante.

gr_erase(Tag) détruit tous les objets étiquetés par Tag de la fenêtre graphique courante.

gr_drawtree(T, R) dessine l'arbre T dans le rectangle (partiellement inconnu) R, qui est une liste de quatres coordonnées ($[X1, Y1, X2, Y2]$).

9.12 Quelques Problèmes et limitations

La version de TCL/TK utilisée pour faire fonctionner l'environnement graphique de Prolog IV n'étant pas la dernière, d'anciens petits problèmes liés à cet outil sont toujours présents.

La prochaine version de Prolog IV utilisera une version récente de TCL/TK et remédiera à la plupart d'entre eux.

– Dans certaines conditions, remettre une fenêtre au premier plan peut prendre quelques secondes (c'est corrigé dans la dernière version de TK, non encore intégrée).

– Pour accéder aux accélérateurs de menu (les lettres soulignées, pas les raccourcis!) il faut que le focus soit dans la partie texte, autrement dit y avoir cliqué.

– La plupart des problèmes de focus sont réglés dans la dernière version de TK.

– La coexistence des deux modes de sélection Macintosh et X11 rend les choses un peu confuses. (la dernière version de TK rend les choses plus claires).

– La coloration syntaxique (dynamique ou pas) peut prendre un certain temps.

– Toute l'interface utilisateur est en anglais.

– Il n'est pas possible de faire des Recherche/Remplacement en arrière.

– Il n'est pas possible de faire des Recherche/Remplacement de texte contenant un retour-chariot. On est donc limité à une seule ligne de texte à rechercher.

– Les colorations syntaxiques des modes C/C++ et TCL ne sont pas implantées.

– Une lettre accentuée est considérée comme un mot isolé par l'éditeur.

– Il faudrait fournir plusieurs jeux de styles pour la coloration syntaxique en fonction du nombre de couleurs de l'écran (pour l'instant, on est supposé avoir un écran couleur, ou noir et blanc)

– Il faudrait également permettre lédition de ces styles (un programmeur connaissant TCL/TK peut le faire).

– Pas de « Undo » dans l'éditeur.

– Alors que Prolog IV est conçu pour travailler avec le jeu de caractère de la machine, le cas du NeXT présente une particularité intéressante :

  – sous shell, Prolog IV doit utiliser le jeu NeXTStep.

  – sous l'environnement graphique, Prolog IV doit utiliser le jeu ISO (X11).

  Bref, Prolog IV ne fonctionne parfaitement avec les accents que dans une fenêtre Terminal. (Il faudrait une option de la ligne de commande pour choisir dynamiquement le jeu de caractère, ou bien encore les classes de caractères pour les entités lexicales)

– Des items de menus ne sont pas encore implantés (ils sont généralement grisés et commencent par un « ? »).

– Dans certains cas, le source montré dans la fenêtre du débogueur n'est pas le bon (alors que ce qui est montré dans le modèle des boîtes l'est toujours à certains numéros près.

Cette version de l'environnement graphique de Prolog IV est un premier jet et s'avère déjà à l'usage d'utilisation très agréable. La correction de la plupart des problèmes, ainsi que de nombreuses possibilités seront disponibles dès la prochaine version.