

EUROPEAN ORGANISATION FOR NUCLEAR RESEARCH

PS/CO/Note 81-6

9.2.1981

| REVISION 8.4.1981 |

NOUVEAUX OUTILS SOFTWARE DES CONSOLES

F. Perriollat
A. Gagnaire

(text formatting by REPORT program)

Table des Matières

| | |
|---|---|
| 1. Relecture des légendes des boutons du "touch-panel user" | 1 |
| 2. Lecture d'une ligne de texte d'un écran vidéo | 2 |
| 3. Message d'erreur du système graphique des consoles | 3 |
| 4. Numéro de groupe pour les "événements machines" | 4 |
| 5. Edition du journal de bord | 4 |

Da=Revision 8.4.1981

De nouvelles facilités ont été introduites pour les programmes d'applications dans les consoles par la distribution Janvier 1981 du software console.

Ces facilités se découpent en deux catégories:

- Des modifications du fonctionnement des triggers software envoyés au MIP afin de rendre le protocole entre le TIP et le MIP plus facile à mettre en oeuvre pour certains cas. Ceci est décrit dans le note PS/CO/Note 81-5 "Les outils software disponibles pour le protocole TIP <--> MIP dans le cas d'édition interactive sur les écrans des consoles".

- De nouvelles fonctions. C'est cette dernière catégorie qui est décrite ci-contre.

1. Relecture des légendes des boutons du "touch-panel user"

Position du problème:

L'organisation du MIP en "overlay" a fait qu'il est apparu comme souhaitable de pouvoir sauvegarder l'état du "touch-panel user" à l'entrée d'un overlay, et le restaurer avant de retourner au moyen du MIP.

La solution que nous avons mise en oeuvre permet d'avoir autant de niveaux d'overlay que l'on veut.

Spécification de la routine:

séquence d'appel (NODAL)

\$SET string = LGREAD(no. de bouton).

appelable uniquement depuis le MIP.

* numéro de bouton dans l'intervalle [1, 16]

- si le bouton est actif: retourne la légende du bouton;

- si le bouton est passif: retourne une chaîne de caractères vide.

* numéro de bouton = 0. Retourne la ligne de titre de la page.

Notes de fonctionnement:

A) Cette facilité ne peut pas relire les "commentaires" écrits directement sur l'écran (par WRITE(LEGLUN)...).

B) Le fonctionnement de cette routine est directement compatible avec l'écriture d'une page complète de légendes par la fonction IGPAGE.

Exemple d'utilisation: (exemple NODAL)

```
2.01 % Sauvegarde de la page courante du TP
2.10 DIM-ST PAGE.
2.12 FOR I = 1,16; $SET PAGE.(I) = LGREAD(I)
2.14 $SET PAGE.(17) = LGREAD(0)

2.20 DO xy; % exécution de l'overlay

2.80 % Restauration de la page
2.82 $SET IGPAGE(PAGE.) = PAGE.(17)

2.90 END; % retour au niveau précédent.
```

2. Lecture d'une ligne de texte d'un écran vidéo

Position du problème:

Certains programmes d'édition prendraient avantage de pouvoir relire le contenu de lignes d'écran pour réorganiser par exemple le texte.

Spécification:

```
séquence d'appel (NODAL)
$SET string = TVREAD(lun, line-number)
```

lun = logical unit number pour l'écran considéré: COLOUR ou BANDW(n).

line-number = position de la ligne à relire; (la ligne no. 1 est la ligne au sommet de l'écran, 24 lignes sont visibles sur un écran).

Retourne dans "string" la chaîne des caractères contenus dans la ligne à relire.

Notes de fonctionnement:

A) Cette facilité est disponible à la fois pour l'écran couleur, ou pour les petits écrans noir et blanc à condition que se soit le générateur de caractère de la console qui lui soit connecté.

Il n'est donc pas possible de relire les textes issus des générateurs de caractères des calculateurs de processus.

B) Le même algorithme d'extraction des caractères que pour l'édition interactive (TVEDIT, TVASK) est utilisé. Ce qui veut entre autre dire que les informations "typographiques" sont perdues: telles que la couleur, les caractères spéciaux définis par l'utilisateur.....

3. Message d'erreur du système graphique des consoles

Position du problème:

Pouvoir disposer directement de message d'erreur en clair pour les "erreurs utilisateurs" du système graphique des consoles. En effet, les erreurs graphiques retournées à l'utilisateur par les fonctions de teste et de synchronisation (GRCHK, GRERR) sont toujours codées.

Spécification:

|| \$SET string = GERMES (error number)

Retourne dans string le message d'erreur correspondant au numéro d'erreur.

Notes de fonctionnement:

A) Un code d'erreur nul (0), qui signifie pas d'erreur, retourne une chaîne de caractères vide.

B) La technique suivante

|| fonction graphique; TY GERMES(GRCHK)

qui supprime le parallélisme d'exécution entre le système graphique et l'utilisateur est absolument déconseillée en dehors de session de "debugging".

4. Numéro de groupe pour les "événements machines"

Un variable "read only" est disponible qui contient le numéro de groupe des événements machines:

EVGRUP

Cette variable permet donc de traiter de façon explicite les groupes au retour d'un "multi-wait" (MWAIT) par exemple.

Note:

La fonction BUTTON qui:

- a) est une "free fonction" donc n'est pas disponible pour les programmes compilés;
- b) exécute une sélection "à priori" des événements, alors que ceux ci devraient être toujours sélectionnés à posteriori.

Ne doit plus être utilisée. Progressivement les appels de cette fonction dans des programmes existants doivent être remplacés par des appels à MWAIT.

Cette fonction BUTTON sera supprimée (avec préavis conserté) dans une prochaine distribution du software console.

5. Edition du journal de bord

Position du problème: L'une des deux imprimantes (imprimante MATRIX) du calculateur MCR est dédiée à la publication automatique du journal de bord.

Une facilité permettant d'écrire une ligne de ce journal de bord doit être disponible facilement à partir de tous les calculateurs du réseau pour les programmes Nodal ou compilé.

Spécification:

NEWS (concatenation)

Ecrit sur le journal de bord une nouvelle ligne contenant:

- la date et l'heure (format ISO)
- deux espaces
- la chaîne de caractères de la "concatenation". Exécute une avance de papier et un retour en début de ligne.

Distribution

PS-CO Listes 7a), 8 extended, 10.