



CLUB INFORMATIQUE DES GRANDES ENTREPRISES FRANÇAISES 21,  
avenue de Messine, 75008 Paris, tél. 256.94.00 poste 24.94

Réf. 76 **116**

Février 1976

PROPOSITION POUR AMELIORER LA  
PORTABILITE  
DES APPLICATIONS INFORMATIQUES

## SOMMAIRE

Préambule

### 1 : ANALYSE DE LA SITUATION ACTUELLE EN MATIERE DE PORTABILITE DES APPLICATIONS

1-1 L'EQUIPE SYSTEME

1-1.1 Formation

1-1.2 Langage de base

1-2 LES APPLICATIONS EN LANGAGES EVOLUES

1-2.1 Cobol

1-2.2 Fortran

1-3 LES FICHIERS

1-3.1 Codification

1-3.2 Support physique

1-3.3 Organisation et accès logique

1-3.3.1 Organisation séquentielle

1-3.3.2 Organisation non séquentielle

1-4 LES LANGAGES DE COMMANDE

1-5 GESTION DES BASES DE DONNEES

1-6 GESTION DE TRANSACTIONS ET RESEAUX DE TELETRAITEMENT

1-6.1 Les problèmes spécifiques du télétraitement

1-6.2 Les niveaux de compatibilité

1-6.3 Etat de la normalisation

1-6.4 Les actions nécessaires

### 2 \_ RECOMMANDATIONS

Conclusion

... j ...

P R E A M B U L E

Le Club Informatique des Grandes Entreprises Françaises, fort des expériences vécues par certains de ses membres qui ont récemment changé de système informatique, a créé un groupe de travail afin d'étudier les conditions économiques et techniques dans lesquelles ces opérations peuvent se dérouler.

L'importance des sommes engagées dans les opérations de transfert d'un système à un autre- conduisent les utilisateurs à examiner les moyens les mieux adaptés pour protéger leurs investissements en matière de logiciels d'applications.

Les dépenses afférentes aux opérations de reconversion sont improductives. Elles dépassent largement les coûts des matériels concernés, et il importe de souligner que la part relative du logiciel d'application dans les investissements informatiques ne fera que croître avec la généralisation des systèmes de gestion de bases de données et de gestion de réseaux téléinformatiques.

Le cadre de cette étude est délibérément celui d'utilisateurs de l'informatique, placés dans le contexte quotidien du traitement de l'information de leur entreprise et sans préoccupations futuristes, c'est-à-dire:

- argumentant leur analyse sur les réalisations informatiques actuelles ou en cours de développement dans leur entreprise
- soucieux de conserver leur entière liberté en matière de choix de constructeurs
- refusant de se voir imposer des systèmes informatiques, de se laisser ~ ~ enfermer dans un ensemble de contraintes résultant de logiciels-de base, de procédures, de réseaux, ...
- et exigeant de participer à la concertation sur les orientations d'une politique des produits qui leur sont destinés.

Les utilisateurs désirent une évolution de l'informatique qui leur offrira demain des possibilités~ qu'ils n'ont pas aujourd'hui, du fait des progrès de la technologie (abaissement des coûts, technologies nouvelles ... ), Ils souhaitent une évolution progressive, à l'image du développement des entreprises, sans remises en cause brutales du fait de leurs fournisseurs, ou de pressions externes, tout en se réservant la possibilité de développer des applications nouvelles.

Au moment où se posent aux pouvoirs publics les problèmes d'orientation d'une informatique nationale, le CIGREF se fait l'interprète de ses membres pour éclairer certains aspects particulièrement importants à ses yeux.

... / ...

## 1 \_ ANALYSE DE LA SITUATION ACTUELLE EN MATIERE DE PORTABILITE DES APPLICATIONS

Les utilisateurs tiennent d'abord à délimiter le domaine qui est le leur, celui des applications.

Les constructeurs développent les matériels et les logiciels de base comme ils le désirent dans un contexte de saine émulation. La seule exigence des utilisateurs est d'obtenir une continuité d'exploitation de leurs applications dans des conditions économiques satisfaisantes.

.../...

## 1-1. L'EQUIPE SYSTEME

La séparation entre informatique des utilisateurs et informatique des fournisseurs n'est pas nette. Aussi les Sociétés membres du CIGREF sont conscientes de la nécessité de plus en plus affirmée d'une interface d'informatique technique dans les entreprises: l'équipe système. Ils sont prêts pour cela à consentir des efforts importants en cas de changement d'équipement, car l'efficacité de cette équipe est un préalable indispensable à un développement cohérent de leur informatique.

### 1-1.1 FORMATION

Les estimations convergentes des entreprises du CIGREF sont les suivantes

- Formation propre des membres de l'équipe système

La reconversion d'un membre confirmé de l'équipe système demande trois mois de cours et un an de rodage au nouveau système. Mais il faut attendre une année supplémentaire avant que cette personne retrouve son efficacité originelle.

- Formation des utilisateurs-informaticiens (chefs de projets ; analystes ; programmeurs; personnel d'ordonnancement, de préparation, de pupitrage) :

Leur indispensable formation doit être assurée par l'équipe système qui a la connaissance des interactions entre exploitation et applications. Il faut en outre insister sur l'importance de la durée de préparation de cette formation qui est de l'ordre de 4 heures pour 1 heure de cours (par exemple, une sensibilisation à un gros système à mémoire virtuelle exige un mois de préparation).

### 1-1.2 LANGAGE DE BASE

Les développements d'applications nouvelles se font en langages évolués; mais dans des cas précis, il est nécessaire de disposer de langages du type assembleur dont les instructions sont liées directement aux instructions-machine. L'utilisation de ces langages est nécessaire pour atteindre des parties du système d'exploitation inaccessibles aux langages évolués afin d'optimiser des parties de traitement coûteuses, de palier les insuffisances des langages évolués et de développer des programmes utilitaires.

Les langages de type assembleur, ou langages de base, n'ont jamais fait l'objet d'une normalisation. Aussi la transposition des programmes qui les utilisent est-elle très coûteuse : il faut pratiquement les réécrire et même adapter l'analyse aux particularités des machines et des systèmes d'exploitation.

Or une exploitation importante peut utiliser environ 100 000 instructions assembleur, et l'on estime le rythme de reconversion à 500 ou 700 instructions par homme-mois.

Certains constructeurs ont développé des traducteurs d'assembleur, dont le degré d'efficacité dépend essentiellement de la structure comparée des instructions (une, deux, trois adresses ; nombre de registres ; tests logiques; fonctions câblées...). Cependant, on observe fréquemment un rapport de 2 à 3 entre le nombre d'instructions engendrées et le nombre

d'instructions d'origine. Par ailleurs, le degré d'efficacité du traducteur n'est pas de 100 % et la partie la plus complexe de la conversion reste à la charge du programmeur.

Les simulateurs et émulateurs ne peuvent être considérés que comme des outils permettant d'étaler dans le temps les charges de reconversion. Leur usage prolongé entraîne rapidement des dépenses importantes liées aux performances et aux problèmes de maintenance des langages-origine abandonnés.

La solution au problème de la portabilité des sous-programmes et des sous-systèmes écrits en assembleur doit être recherchée dans une généralisation de l'utilisation de langages du type CPL ou LPS. Cette généralisation ne serait réellement bénéfique que si de tels langages étaient normalisés (dans leur syntaxe et dans leurs fonctions). Cependant, il restera des cas où il faudra recourir à l'assembleur d'origine (par exemple dans le cas d'intervention au niveau de la structure du système) ; la portabilité sera alors délicate.

En conclusion, l'équipe système sera donc lourdement mise à contribution durant une période minimum de deux ans en cas de changement de constructeur pour les aspects suivants

- formation préalable de l'équipe elle-même,
- préparation des cours et formation des utilisateurs au nouveau système,
- définition des paramètres du nouveau système d'exploitation adaptée à la charge du site,
- prise en charge de la mise en exploitation des applications transposées, -
- traduction des programmes utilitaires et des programmes en langages de base,
  
- optimisation de la nouvelle exploitation,

tout en assurant, pendant une partie de ce temps de mutation, la maintenance du système remplacé.

## 1-2-. LES APPLICATIONS EN LANGAGE.S EVOLUES

Il s'agit là essentiellement des applications programmées en COBOL et en FORTRAN. Les problèmes liés aux langages PL1 et APL n'ont pas été examinés, bien que certains sites importants en fassent grand usage; mais il s'agit d'opérations ponctuelles.

Les problèmes de portabilité des applications programmées en COBOL ou en FORTRAN se posent dans les mêmes termes : des normes définissent ces langages, mais les constructeurs ou Sociétés de services ont ajouté des possibilités non normalisées pour mettre en valeur certains aspects spécifiques de leurs produits. Il en résulte donc des incompatibilités entraînant des coûts de reconversion importants.

### 1-2.1 COBOL

La grande masse des applications de gestion, est aujourd'hui programmée en COBOL. Les problèmes liés à la normalisation de COBOL et à la définition de sous-ensembles portables du COBOL (COBOL ANS, COBOL C ... ) ont été trop souvent exposés dans la presse pour qu'il soit nécessaire d'en parler ici.

Le fait d'imposer aux programmeurs des limitations dans l'utilisation de ce langage pour se conformer aux prescriptions d'un COBOL restreint facilite les transpositions, mais au prix des investissements que représentent :

- la persuasion des programmeurs à limiter les possibilités de leur langage de programmation,
  - le contr8le du respect des règles instaurées,
  - les pertes de performances éventuelles.

Comme pour les langages assembleur, il existe des traducteurs de COBOL, avec les mêmes lacunes: les véritables problèmes de portabilité ne sont pas résolus par ces outils.

Les sommes engagées pour assurer la transposition des programmes COBOL sont importantes; elles dépendent de la complexité des programmes, mais surtout de la compatibilité des systèmes d'exploitation origine et objet. D'un minimum absolu de 8 Frs l'instruction (passage d'un système type DOS vers un système OS), on aboutit à 15-20 Frs l'instruction dans des cas complexes (COBOL non normalisé, système origine évolué) et ce en tenant compte de la formation des programmeurs, des marches en double, de la conversion des fichiers et des langages de commande. Pour un centre moyen dont l'exploitation utiliserait un million d'instructions COBOL, le seul coût de conversion de celles-ci dépasserait le prix d'achat de la machine utilisée. De plus, se poseraient les problèmes liés aux validations et marches en double des applications reconverties, la sécurité des résultats n'étant acquise qu'après un certain nombre d'exploitations. Les erreurs révélées au cours de ces traitements sont particulièrement difficiles à corriger si la conversion a été confiée à un organisme extérieur.

.. . / ...

Des moyens propres à minimiser les couts de transposition des applications COBOL peuvent être cherchés dans les voies suivantes :

- \_ Normalisation plus poussée du COBOL avec, en premier lieu mise au point de pré compilateurs détectant toute liberté prise avec la norme.
  
- Utilisation éventuelle de méthodes d'analyse ou de générateurs de COBOL. Mais les utilisateurs sont loin d'être convaincus des avantages qu'ils présentent. Par ailleurs, une telle démarche supposerait que l'on garantisse le développement de tous ces produits sur tous les matériels nouveaux.

### 1-2.2 FORTRAN

On rencontre dans la transposition des programmes FORTRAN les mêmes problèmes qu'en COBOL; malgré l'existence de normes, les compilateurs renferment des différences notables dues aux développements apportés par les constructeurs.

On trouve cependant un certain nombre de problèmes spécifiques

- Longueur variable des mots: d'où problème de précision (passage d'un système à mot de 60 bits voire 36 bits à un système à mot de 32 bits).
  
- \_ Fonctions mathématiques conduisant à des résultats distincts du fait de choix d'algorithmes différents (exponentielles ... ).
  
- Modes de gestion des recouvrements -{internes ou externes.

Toutes ces difficultés peuvent nécessiter une intervention importante portant sur la structure même des programmes. On y ajoutera celles liées à la nature des dits programmes : peu ou pas de dossiers d'analyse-programmation, programmes non figés conduisant à des modifications fréquentes.

On ne peut imposer une longueur de mot à un constructeur ; cette impossibilité constitue un frein majeur~ une portabilité complète.\_

Les actions possibles au niveau de ce langage sont donc limitées.

Tout au plus, peut-on souhaiter voire normaliser le langage et la façon de gérer les recouvrements.



### 1-3. LES FICHIERS

L'informatique traite des données, lesquelles sont codées et organisées sur des supports physiques auxquels il faut accéder. Les difficultés de portabilité de fichiers proviennent des points suivants:

- Codification
- Support physique
- Organisation et accès logique.

#### 1- 3. CODIFICATION

Les problèmes de différence de codification (codes à 6 ou 8 bits, codes de perforation des cartes et rubans) peuvent être résolus par des convertisseurs mais cette situation reste irritante pour les utilisateurs qui souhaitent s'affranchir de ces contraintes ~ souvent artificielles imposées par les matériels.

#### 1-3.2 SUPPORT PHYSIQUE

La normalisation des supports physiques n'est assez avancée et respectée que dans les domaines suivants :

- Carte perforée 80 colonnes
- Bande magnétique.

Des progrès ont aussi été réalisés en matière de disques magnétiques, mais des divergences subsistent.

#### 1-3.3 ORGANISATION ET ACCÈS LOGIQUE

Les vrais problèmes se posent lorsqu'on veut accéder aux données d'un fichier créé sur un système différent de celui de l'exploitation. On se limite ci-après-aux méthodes habituelles à l'organisation des fichiers magnétiques :

- séquentielle,
- non séquentielle.

#### 1-3.31 Organisation séquentielle

Sur l'habituel support bande, sous réserve des problèmes posés par les labels, une bonne portabilité existe (du moins sur 800 et 1600 bpi). Une normalisation des labels plus poussée est cependant **indispensable**. Le développement d'une unité bi-densité pour les définitions plus **fin**es facilitera aussi les conversions de fichiers.

.../...

### 3.32 Organisation non séquentielle

La portabilité est pratiquement nulle car les problèmes se posent en terme d'implantation différente des tables, des catalogues, de gestion des débordements ...

Une solution à ces problèmes, vitale pour toute entreprise en cours de changement de système informatique ou qui a besoin de transporter en période d'exploitation des fichiers d'un système sur l'autre, pourrait, pour des fichiers qui ne sont pas gigantesques, passer par la définition d'un Fichier Interface Standard (F.I.S.). Ce **F.I.S.** serait d'organisation séquentielle. Il comporterait un label normalisé comportant toutes les informations nécessaires à son implantation sur tout autre système. Pour chaque constructeur et chaque système, il y aurait donc lieu de développer les produits réciproques d'interface:

machine origine - fichier interface

fichier interface - machine objet.

Il ne s'agirait en fait que d'une généralisation des opérations de sauvegarde et restauration.

. . . / . . .

#### 1-4. LES LANGAGES DE COMMANDE

Il ne s'agit de traiter ici que les aspects concernant strictement l'utilisateur  
description des fichiers, accès à ceux-ci,

- lancement et enchaînement de travaux avec gestion de "codes retour"  
pour les enchaînements conditionnels.

Ce problème est simple en lui-même mais nécessite une réécriture longue,  
fastidieuse et génératrice d'incidents parfois graves en cas d'erreurs dans  
la prise en compte des codes-retour.

La solution à ce problème peut être apportée par la définition et la  
normalisation d'un "langage de commande évolué" traduit en langage de commande  
normal par un compilateur et s'appuyant sur des fonctions implicites définies  
à la génération du système de base ou au chargement. En effet, il ne s'agit  
là que d'indiquer des relations simples (entre fichiers, supports physiques  
de ces fichiers, programmes) qui sont fonctionnellement identiques pour tous  
les systèmes mais formulées très différemment car les "JCL" sont très orientés  
machine, ce qui est, semble-t-il obligatoire.

### 1-5. GESTION DES BASES DE DONNEES

La mise en œuvre et la gestion des bases de données ne sont pas actuellement maîtrisées ; leur utilisation, aujourd'hui limitée, va se développer rapidement car elles apportent aux utilisateurs des perspectives intéressantes.

- de structuration logique des données, indépendamment des traitements ultérieurs,
- d'accès parallèle à ces données par plusieurs applications, sans préoccupation de leur organisation physique,

tout en offrant les garanties liées à la saisie unique des données, ainsi qu'à leur sécurité et leur fiabilité. L'utilisateur final est en outre dégagé d'un certain nombre de contraintes fonctionnelles qui sont alors du domaine du "gestionnaire de la base" ou traitées par une fonction de "structuration de la base". La portabilité des bases de données est donc un problème important étant donné le développement attendu dans ce domaine, l'ampleur des investissements nécessaires et l'impact qu'elles auront sur la vie des entreprises.

On ne dispose pas de données significatives pour parler de reconversion, mais on peut considérer que la portabilité en la matière est pratiquement nulle, à deux exceptions près :

- \_ SOCRATE qui pour un seul concept dispose de trois réalisations différentes : version IBM et versions SIRIS 3 et SIRIS 8 sur matériel CII (mais ces réalisations sont encore loin d'être transparentes)
- SYSTEM 2000 avec des versions IBM, CDC et UNIVAC.

Une normalisation est vitale et urgente. Elle est commencée au sein du DBTG (Data Base Task Group) et de la commission CODASYL qui s'attachent à définir les spécifications auxquelles devrait satisfaire un système de gestion de base de données. Dans le cadre de l'AFNOR, le problème est aussi examiné. Mais au plan international, IBM semble actuellement se désintéresser des recommandations de ces comités.

Cette normalisation devrait d'abord définir le concept même de base de données dans l'esprit suivant :

- \_ structuration logique des données et de leurs relations tenant compte des systèmes existants ;
- \_ définition des langages de structuration, chargement, validation de la base ;
- \_ manipulation des données, avec définition des interfaces (langages évolués, systèmes de gestion de transactions).

Toute réalisation informatique de ce type devrait pouvoir se faire dans un langage d'un type évoqué précédemment (LPS ou CPL **1**).

. . . / . . .

## 1-6. GESTION DE TRANSACTIONS ET RESEAUX DE TRANSMISSION

### 1-6.1 LES PROBLEMES SPECIFIQUES DU TELETRAITEMENT

Un changement de fournisseur de produits informatiques pose des problèmes spécifiques dans le cas de systèmes comprenant un important réseau de transmission de données, du fait de facteurs tels que le grand nombre de terminaux, leur dispersion géographique, et le grand nombre de personnes en contact avec le système informatique (employés de guichet, agents administratifs, etc ...). En particulier, une augmentation de la fréquence des pannes du système central ou une dégradation des performances (temps de réponse) qui se produisent souvent lors d'une reconversion, sont beaucoup plus ressenties que dans les systèmes où le traitement de l'information est localisé entre les mains de spécialistes toutes causes pouvant entraîner des perturbations dans le fonctionnement de l'entreprise.

Une marche en double du système, outre son coût important, n'est en général pas possible pour des raisons de logistique au niveau des terminaux. Dès qu'un réseau de télétraitement dépasse une certaine importance quantitative (une dizaine de terminaux) on ne peut reconvertir en même temps l'ensemble du système (ordinateur central et terminaux).

Un changement de l'ordinateur central doit dès lors pouvoir s'effectuer sans modifier les terminaux.

D'une manière générale, dans les réseaux de télétraitement il est toujours préférable que l'évolution se fasse par changements partiels plutôt que par remise en cause de l'ensemble. La portabilité dans ce domaine est très liée à l'adaptabilité, et on analysera ici ces deux aspects.

La possibilité de reconversion partielle (par exemple ordinateur central seulement nécessite qu'il y ait compatibilité entre les ordinateurs centraux et les terminaux de constructeurs différents, et que les programmes d'application et les systèmes opératoires soient indépendants de l'architecture du réseau.

L'indépendance entre les différents composants du système (ordinateur central, réseau de transmission de données, terminaux) est souhaitable pour rendre l'utilisateur indépendant par rapport au constructeur. Il importe en effet que l'utilisateur, au lieu de voir ses choix imposés par un constructeur unique auquel il se serait confié, soit libre

- lors de la conception du système, de choisir pour chacun de ses composants le fournisseur le plus intéressant,
- lors de l'évolution et des extensions ultérieures, de faire appel, si nécessaire, à d'autres constructeurs pour profiter de leurs innovations.

... / ...

### 1-6.2 LES NIVEAUX DE COMPATIBILITE

On s'accorde généralement à distinguer les niveaux de compatibilité suivants :

- Niveau "procédure de transmission"  
c'est celui de l'ensemble des règles permettant d'échanger des données, sans perte et sans erreur, et avec un rendement satisfaisant, sur une ligne de transmission.
- Niveau "protocole de communication"  
qui est celui de l'ensemble des règles permettant à deux équipements informatiques {ordinateurs ou terminaux) de communiquer à distance en synchronisant leurs états, en contrôlant le flux de données, et en réinitialisant la communication après les incidents.
- Niveau "fonctionnement des terminaux"  
c'est celui de l'ensemble des commandes que l'ordinateur central envoie pour déclencher des actions précises sur le terminal. Actuellement, ces commandes varient selon les constructeurs de produits informatiques.
- Niveau "langages d'utilisation du réseau"  
les problèmes posés à ce niveau apparaissent par exemple pour les systèmes de soumission de travaux à distance ainsi que pour les systèmes de gestion de transaction en temps réel.

Dans les systèmes actuellement opérationnels, les différents niveaux de compatibilité ne se correspondent en général pas d'un constructeur à l'autre. Les réseaux de commutation de paquets, tels CIGALE ou TRANSPAC imposent aux utilisateurs qui s'y connectent d'être compatibles avec eux au niveau de la procédure de transmission. Mais ces réseaux n'offrent actuellement pas de solution technique pour réaliser de manière simple la compatibilité aux autres niveaux, et cela parce qu'ils se bornent à remplir la fonction de transmission de données d'un point géographique à un autre, sans chercher à savoir si ces données seront correctement interprétées à chaque extrémité.

Il est d'ailleurs souhaitable que ces réseaux n'offrent que cette fonction, car c'est ce qui rend leur utilisation générale. Les autres fonctions de communication doivent rester séparées du fait des problèmes spécifiques qu'elles posent.

### 1-6.3 ETAT DE LA NORMALISATION

Une normalisation des procédures de transmission (HDLC) est en cours. Peu de constructeurs s'y conforment pour le moment même si certains s'en rapprochent, tel IBM avec SDLC par exemple.

Des efforts sont faits pour normaliser les protocoles de communication, mais les travaux n'en sont encore que dans une phase exploratoire.

Si même elle est possible, une normalisation des niveaux "fonctionnement des terminaux" et "langages d'utilisation du réseau", paraît encore plus éloignée.

... / ...

#### 1-6.4 LES ACTIONS NECESSAIRES

Si on veut s'orienter à terme vers une situation où il soit possible de changer un composant du système de télétraitement sans modifier les autres (l'ordinateur central, une partie du réseau, ou une partie des terminaux) il faut remplir les conditions suivantes:

- a) Les constructeurs de produits informatiques doivent respecter les normalisations lorsqu'elles existent.
- b) Il faut s'efforcer de dégager des normes dans tous les domaines où cela est possible.
- c) S'il n'est pas possible à court terme de définir des normes à tous les niveaux, il est nécessaire que :
  - dans les produits informatiques, les niveaux mentionnés ci-dessus soient bien séparés, les séparations entre ces niveaux coïncident avec les interfaces entre les différents produits constituant le système de télétraitement, chaque produit réalisant une fonction élémentaire soit autonome et dépende le moins possible des autres, les interfaces, même si elles ne correspondent pas à des normes, soient stables, clairement définies et bien documentées de manière à constituer des points d'ouverture où l'on puisse connecter des produits d'autres constructeurs avec un effort raisonnable.
  - d) Les utilisateurs doivent, dans la mesure où les logiciels de base disponibles le permettent, concevoir les programmes d'application de manière à rendre la fonction de traitement des données indépendante des fonctions de transmission et de communication, et en particulier, indépendante des techniques de transmission (procédures, protocoles de la topologie du réseau, de la nature des terminaux.
  - e) On peut envisager la réalisation d'une famille de moniteurs de gestion de transactions compatibles. Chacun pourra fonctionner dans un ordinateur de traitement différent, mais
    - \_ ils présenteront les mêmes interfaces pour les programmes d'application de l'utilisateur écrits dans un langage évolué tel que le COBOL ils admettront la même organisation du découpage des séquences de traitement et de l'enchaînement de ces séquences, le même langage d'utilisation du réseau, etc ...
    - ils seront écrits en langage du type CPL 1 ou LPS.

.../...

## 2 RECOMMANDATIONS

Pour remédier aux faiblesses et répondre aux critiques de la situation actuelle, le C.I.G.R.E.F. est conscient de la nécessité d'une action coordonnée des parties prenantes

- Pouvoirs Publics,
- Constructeurs,
- Sociétés de services en informatique,
- Utilisateurs.

A l'issue de cette réflexion, il propose un certain nombre d'actions qu'il juge très importantes.

.../...



Les répercussions considérables entraînées par des changements d'ordinateurs accompagnés d'un changement de ligne de matériel amène les entreprises du CIGREF à proposer l'établissement de certaines normes et outils.

Ils sont indiqués ci-dessous en trois classes selon un degré de faisabilité décroissante, et, à l'intérieur de ces classes, par ordre de priorité décroissante.

#### Première classe

- Etablissement d'une norme française du COBOL et fourniture du précompilateur associé (cf. t-2.1 ).
- Normalisation des labels de bandes (cf. 1-3.31).
- Définition d'une "trame" standard pour l'établissement d'un Fichier Interface Standard de transfert des fichiers disques (cf. 1-3.32).

#### Seconde classe

- Définition et normalisation d'un langage de commande évolué (cf. 1-4.), fourniture des générateurs de JCL correspondants pour les principaux modèles d'ordinateurs.
- Définition "officielle" du découpage des fonctions de transmission en niveaux - Incitation des fournisseurs de logiciel pour qu'ils respectent ces découpages - Accélération de la sortie de la norme concernant la procédure de transmission (cf. 1-6.2,3 et 4).
- Normalisation des bases de données (organisation, langage d'accès) (cf. 1-5).

#### Troisième classe

- Etude et normalisation d'un langage type CPL I ou LPS (cf. 1 -1 . 2 ) .
- Normalisation du FORTRAN avec, comme pour COBOL, fourniture du précompilateur associé (cf. 1-2.2).

... / ...

## CONCLUSION

Le cout d'une opération de reconversion est toujours très élevé, et il peut excéder celui du matériel informatique. Cela impose d'assurer:

- \_ une évolution progressive des applications et de leur exploitation, par stades intermédiaires transparents aux utilisateurs,
- \_ la protection des investissements en logiciels consentis par les entreprises.

La mise en œuvre d'une nouvelle politique industrielle risque d'engendrer une rupture de la continuité de la politique informatique. Il paraît inconcevable dans ces conditions de faire supporter aux utilisateurs à la fois les couts et les contraintes des changements de cap imposés.

Il importe de permettre la meilleure participation possible des entreprises au succès de la politique industrielle définie par les Pouvoirs Publics, en faisant participer les utilisateurs à la définition de la politique informatique nationale.

Cette politique informatique, construite en particulier grâce à des actions poussées de normalisation et de concertation, devra permettre aux entreprises d'envisager à l'avenir le changement des matériels informatiques dans les mêmes termes qu'un changement de tout autre type de matériel.

Les actions à entreprendre à ce sujet concernent :

- \_ le court terme (entre 1 et 5 ans) : il n'existe pour ce terme guère de solution en dehors de la réalisation d'interfaces entre le logiciel du système de départ et celui du système d'accueil;  
le long terme (au-delà de 5 ans, correspondant à des migrations vers des gammes de matériels non encore annoncés) : ce délai permet la définition concertée entre toutes les parties prenantes d'une politique informatique assurant une bonne portabilité des applications et garantissant les investissements en logiciel des entreprises.

Ces actions devront respecter un certain nombre de principes nettement affirmés :

- \_ celui de la possibilité de choix qui implique l'existence de systèmes concurrents assurant les mêmes fonctions ;
- \_ celui de la possibilité d'évolution de l'informatique des utilisateurs dans des conditions économiques prévisibles et acceptables;
- \_ celui de la concertation nécessaire qui garantira aux constructeurs et Sociétés de services la rentabilisation de leurs investissements qui répondent aux besoins réels des utilisateurs.

L'ampleur des problèmes soulevés et l'importance des intérêts mis en jeu exigent non seulement une participation des utilisateurs, mais surtout de leur part, un effort de concertation et de cohésion. La définition de leurs besoins ne peut être le fait d'aucun d'entre eux pris séparément, quelle que soit sa compétence: elle doit être celui de groupements d'utilisateurs.

Pour sa part, le CIGREF est prêt à participer à la définition des actions de concertation et de normalisation qui permettraient la mise au point d'une politique informatique nationale, lequel requiert, sous l'impulsion des Pouvoirs Publics la participation des constructeurs, des sociétés de services et de s utilisateurs.