

# Interface Homme Machine

1. Introduction
2. Définitions
- 3. Critères ergonomiques**
4. Les étapes du processus de développement et IHM
5. Modèles de tâches
6. Architecture logicielle des systèmes interactifs
7. Les formalismes de validation des DHMs (Dialogues Homme Machine)

### 3. Critères ergonomiques

Les critères ergonomiques permettent de mesurer la qualité de l'interface, selon plusieurs approches ces critères ont été groupés en catégories et classes :

#### → Critères de Scapin et Bastien :

Selon ces chercheurs (INRIA) les critères ergonomiques sont regroupés en 08 catégories :

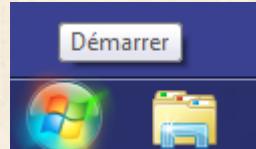
##### a) Guidage

Cette catégorie représente l'ensemble de moyens mis en œuvre pour orienter l'utilisateur lors de ses interactions. *Un bon guidage facilite l'apprentissage et l'utilisation du système.*

Cette catégorie se décompose en 04 critères :

**a.1) Incitation:** il recouvre les moyens mis en œuvre pour amener les utilisateurs à effectuer des actions spécifiques.

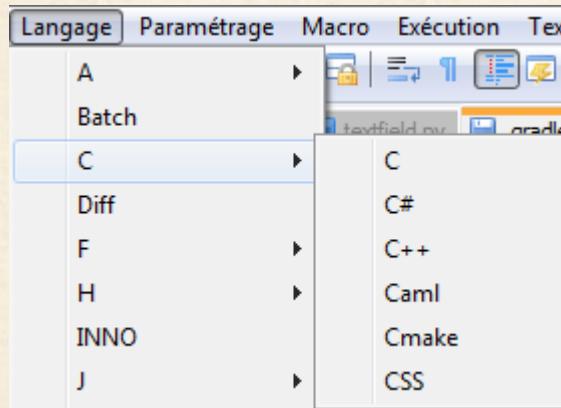
Exemples:



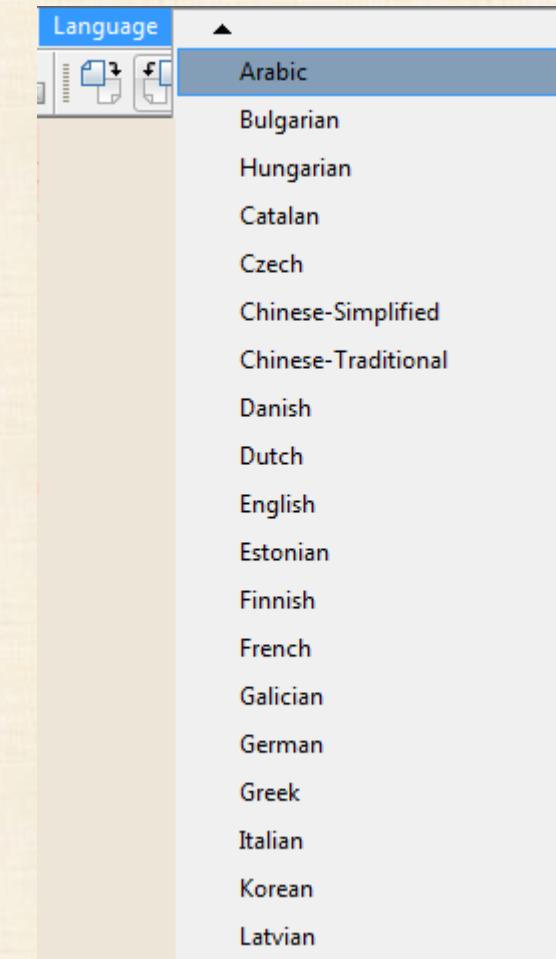
### 3. Critères ergonomiques (Cont. Scapin et Bastien-Guidage)

**a.2) Groupement / distinction entre items :** Ce critère prend en compte la localisation et le format afin d'illustrer les relations entre les items affichés.

Exemple:



Contre Exemple:

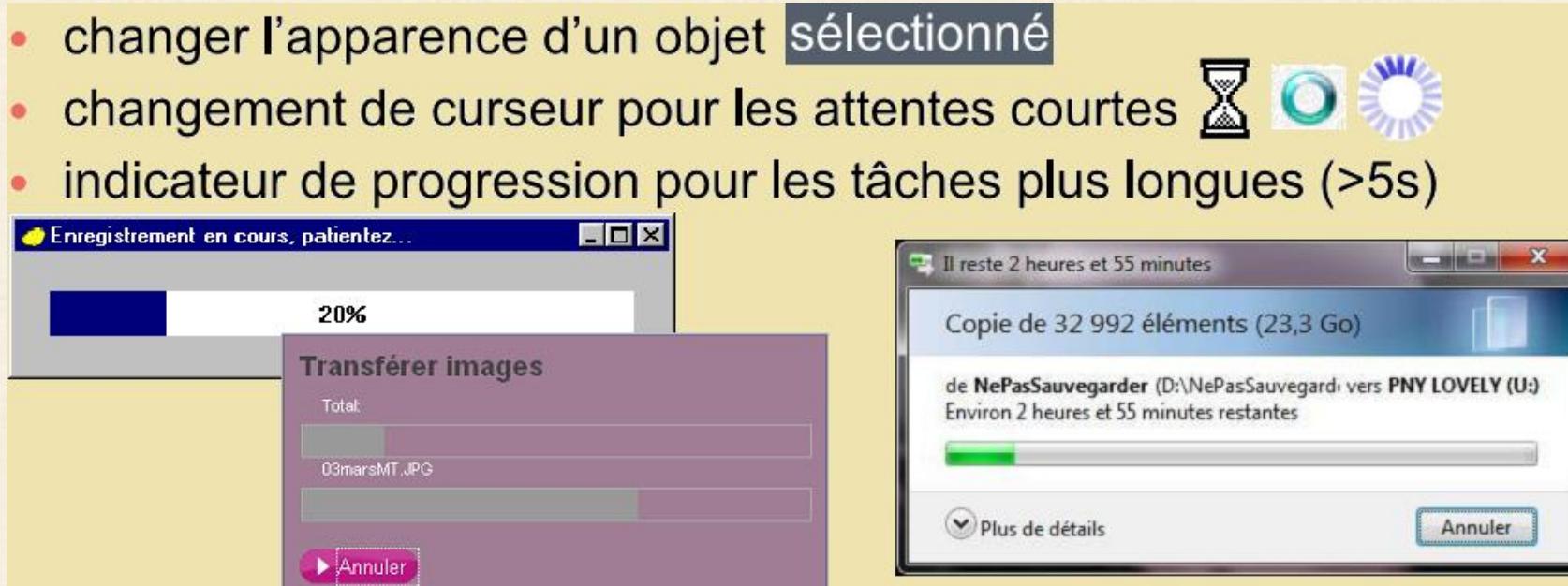


### 3. Critères ergonomiques (Cont. Scapin et Bastien- Guidage)

**a.3) Feedback immédiat** : Il représente la réponse immédiate qui doit être fournie à l'utilisateur.

Exemple:

- changer l'apparence d'un objet **sélectionné**
- changement de curseur pour les attentes courtes   
- indicateur de progression pour les tâches plus longues (>5s)



Contre Exemple:



### 3. Critères ergonomiques (Cont. Scapin et Bastien-Guidage)

**a.4) Lisibilité :** il concerne les caractéristiques de présentation pouvant entraver ou faciliter la lecture de l'interface.

Exemple:

Ergonomie:  
vise à adapter les postes de travail  
aux caractéristiques physiologiques et  
psychologiques de la personne

Contre Exemple:

**Ergonomie logicielle:**

- vise à adapter les logiciels à l'utilisateur
- pour diminuer les erreurs, le temps d'apprentissage
- rendre le logiciel le plus facilement utilisable

### 3. Critères ergonomiques (Cont. Scapin et Bastien- Charge de travail)

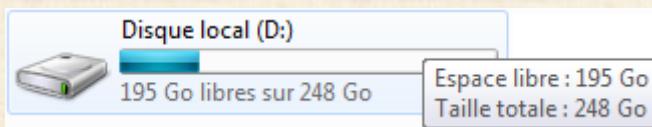
#### b) Charge de travail

Elle représente un ensemble d'éléments de l'interface participant à la réduction de la charge perceptive ou mnésique des utilisateurs. Plus la charge de travail est élevée, plus les risques d'erreurs sont élevés. Cette catégorie comporte deux critères :

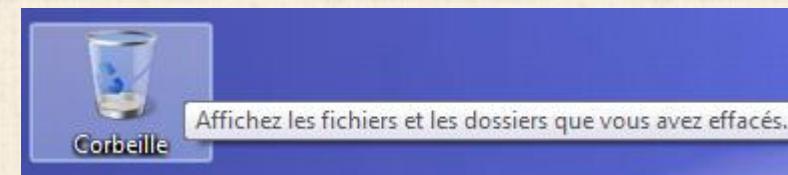
**b.1) Brièveté** : il consiste à limiter autant que possible le travail de lecture, d'entrée, et les étapes par lesquelles doivent passer les utilisateurs (i.e., augmentation de l'efficacité du dialogue) . Ce critère lui-même se décompose en deux sous-critères :

**b.1.1) Concision** : éviter les phrases longues et inutiles pendant les notifications.

Exemple:



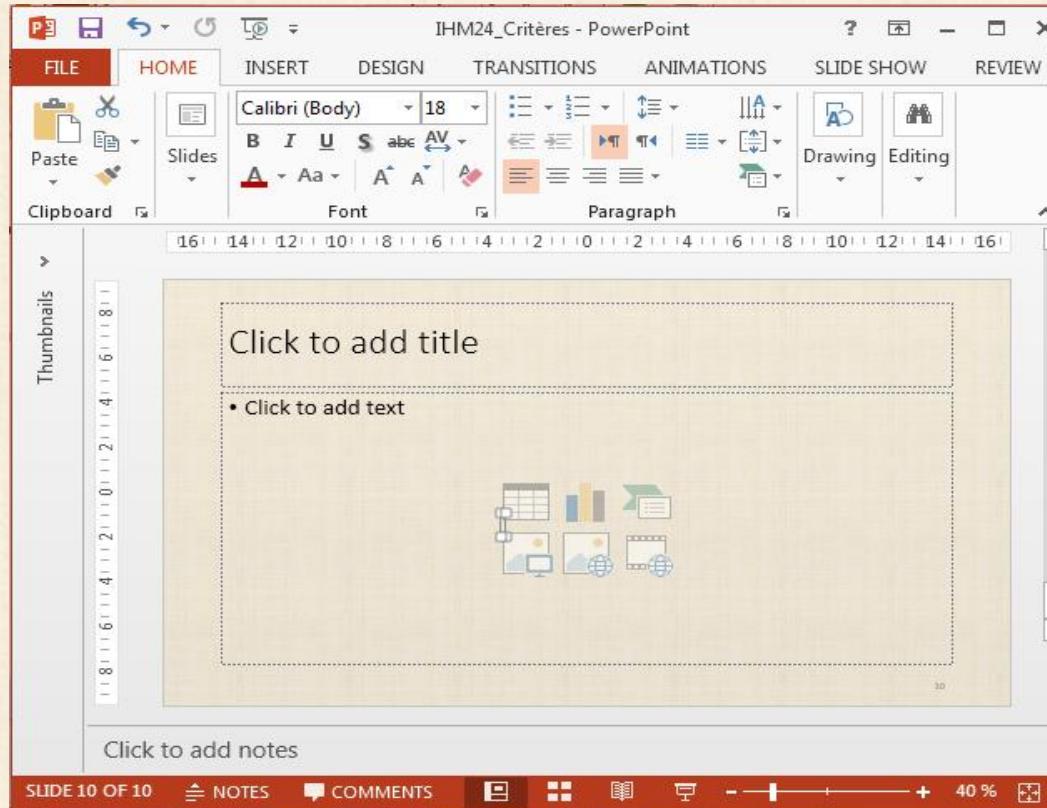
Contre Exemple:



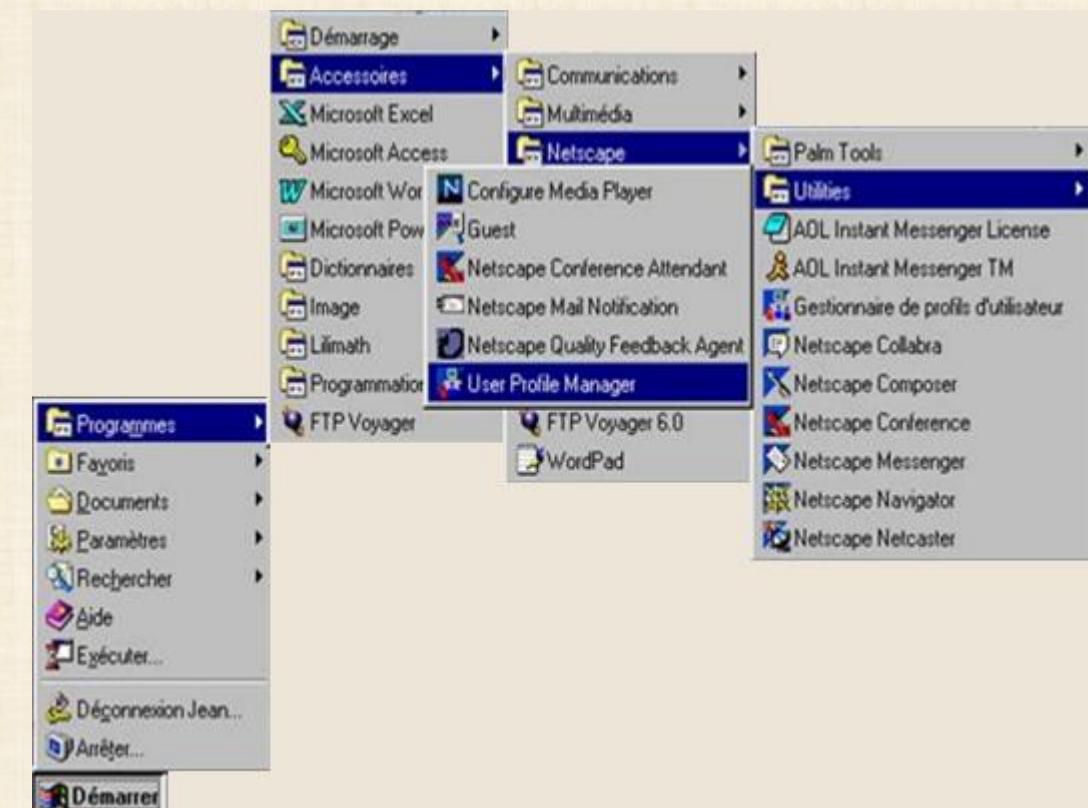
### 3. Critères ergonomiques (Cont. Scapin et Bastien- Charge de travail )

**b.1.2) Actions minimales:** Utiliser les rubans au lieu des menus.

Exemple:



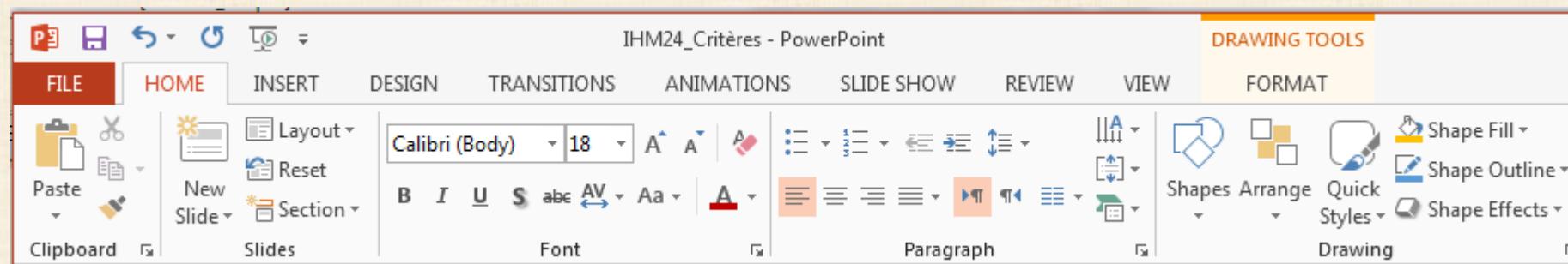
Contre Exemple:



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Charge de travail )

**b.2) Densité informationnelle:** il représente la charge de travail pour des ensembles d'éléments. On doit présenter seulement les éléments fréquemment utilisés.

Exemple:



Contre Exemple:



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Contrôle explicite)

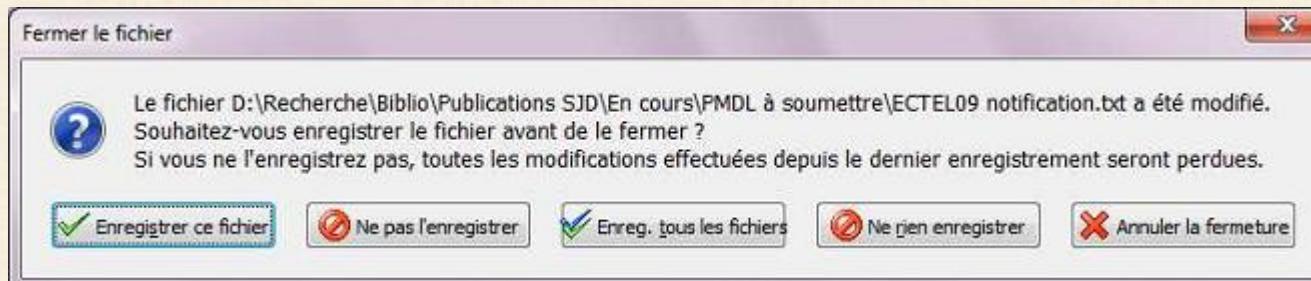
#### c) Contrôle explicite

Elle représente la prise en compte par le système des actions explicites de l'utilisateur. Plus les entrées des utilisateurs sont explicitement définies par eux-mêmes, plus les ambiguïtés et les erreurs sont limitées.

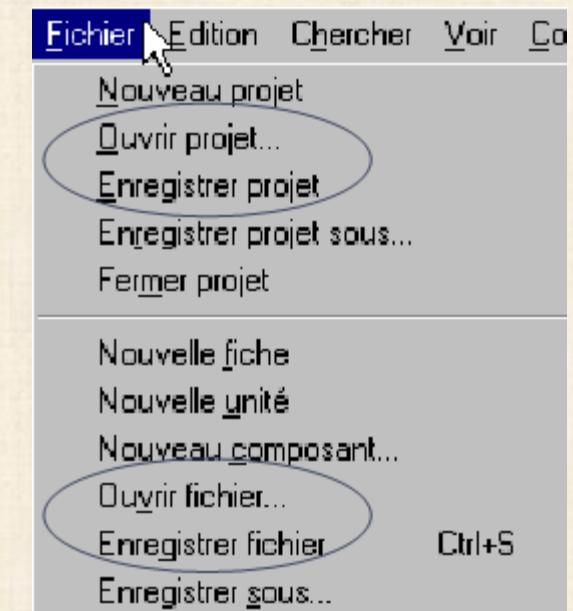
Deux critères pour cette catégorie :

**c.1) Actions explicites :** le système doit prendre en compte les actions souhaitées de l'utilisateur au moment où il les demande.

Exemple:



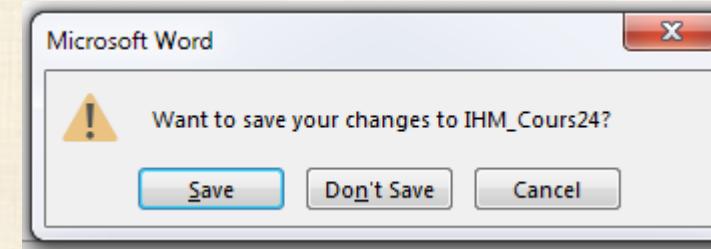
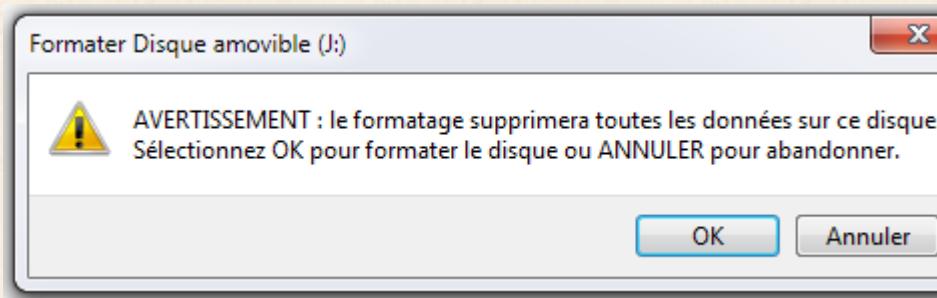
Contre Exemple:



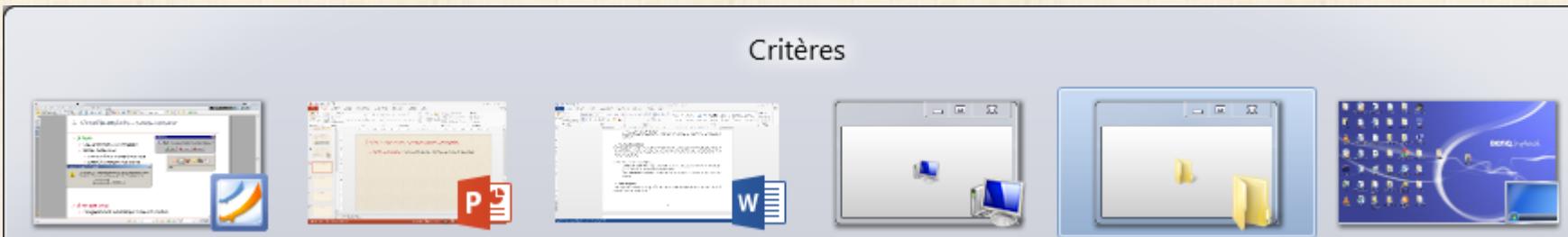
### 3. Critères ergonomiques (Cont. Scapin et Bastien- Contrôle explicite)

**c.2) Contrôle de l'utilisateur :** l'utilisateur doit toujours avoir la main pour contrôler ses activités, Ainsi toujours laisser l'initiative à l'utilisateur: laisser le se déplacer entre ses activités, les interrompre et les arrêter.

Exemple:



Contre Exemple:  
Quitter sans  
enregistrer



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Adaptabilité)

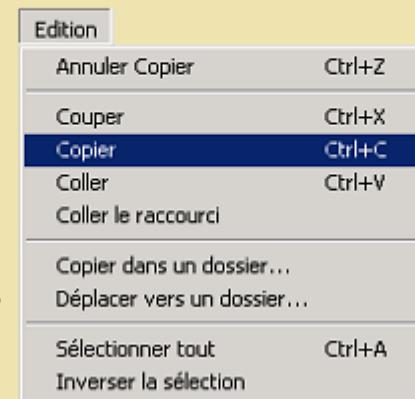
#### d) Adaptabilité

Elle exprime la capacité de l'application à réagir selon le contexte. Une interface ne peut convenir à la fois à tous ses utilisateurs potentiels.

**d.1) Flexibilité** : capacité de l'interface de s'adapter aux actions variées des utilisateurs, plusieurs façons pour exécuter la même action.

Exemple:

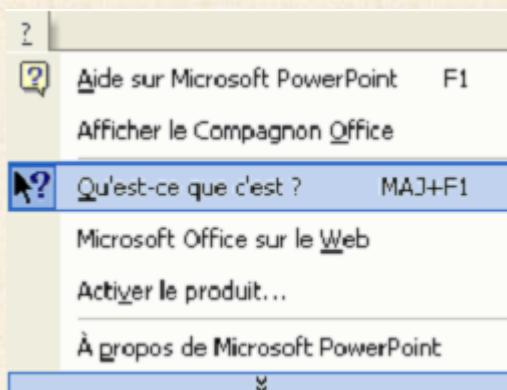
- copie de fichiers vers un support externe dans l'explorateur
  - glisser - déposer
  - copier - coller
    - menus
      - édition
      - contextuel
      - envoyer vers
    - raccourcis clavier
    - boutons 



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Adaptabilité)

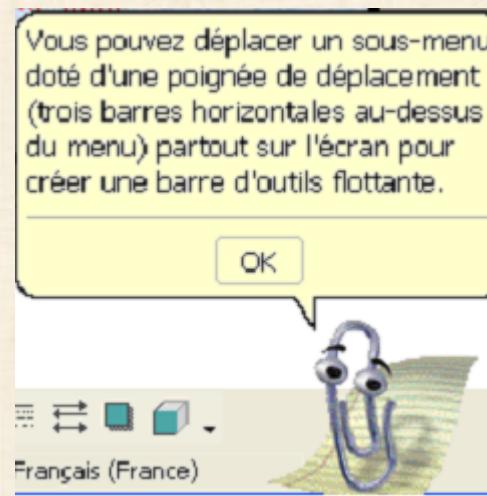
**d.2) Prise en compte de l'expérience de l'utilisateur :** respecter le niveau d'expérience de l'utilisateur.

Exemple: **Simple user**



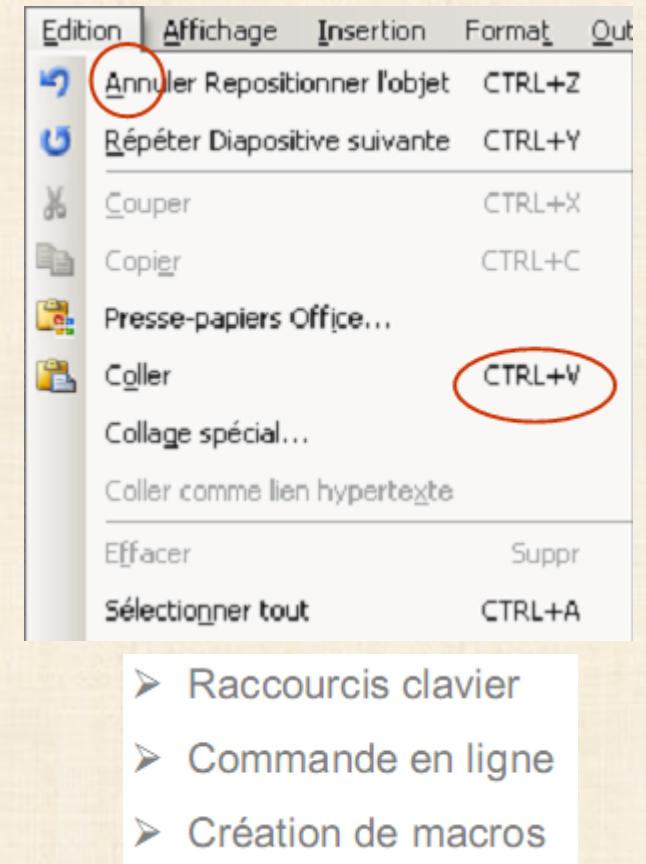
- Limiter le nombre d'actions
- Feedback d'information
- Guidage (exemple : menus)
- Messages d'erreur informatifs
- Manuel d'utilisation intuitif
- Aide contextuelle

**Occasionnel**



- Consistance de l'interface
- Prévention des erreurs
- Aide en ligne

**expert**



- Raccourcis clavier
- Commande en ligne
- Création de macros

### 3. Critères ergonomiques (Cont. Scapin et Bastien- Gestion des erreurs)

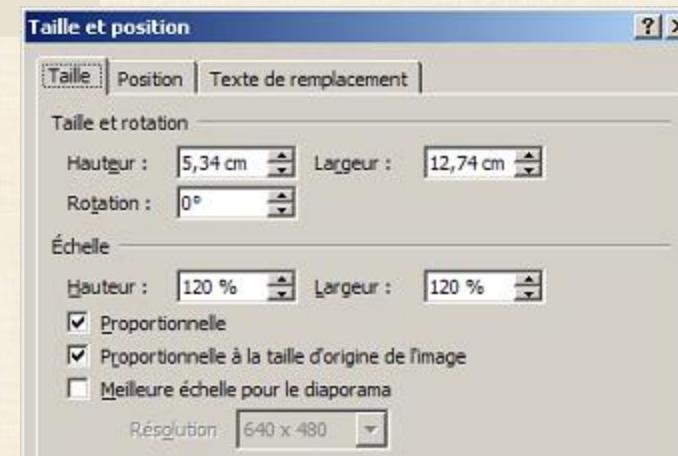
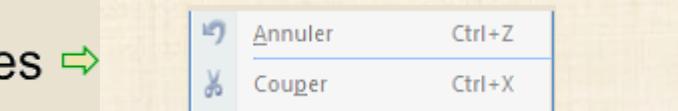
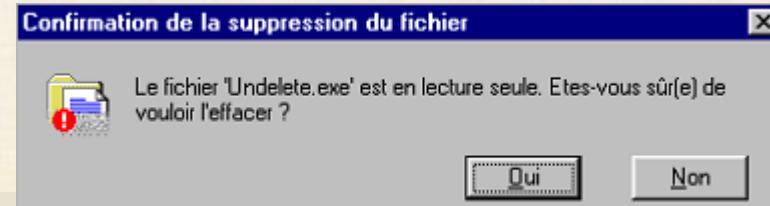
#### e) Gestion des erreurs

Elle s'intéresse aux moyens permettant d'une part d'éviter ou de réduire les erreurs, et d'autre part de les corriger lorsqu'elles surviennent. Les interruptions provoquées par les erreurs ont des conséquences négatives sur l'activité des utilisateurs.

**e.1) Protection contre les erreurs :** détecter et prévenir les erreurs d'entrée de données ou de commandes.

Exemple:

- protéger les actions sensibles ↗
- griser les commandes indisponibles ↗
- fournir des valeurs attendues ↗



### 3. Critères ergonomiques (Cont. Scapin et Bastien- **Gestion des erreurs**)

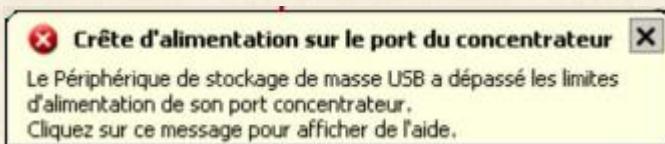
**e.2) Qualité des messages d'erreur :** concerne la pertinence, la facilité de lecture et l'exactitude de l'information sur la nature des erreurs commises.

Exemple:

- messages informatifs, constructifs, compréhensibles, amicaux, montrant la cause et la solution
- 4 types de messages
  - information
  - avertissement
  - erreur bloquante
  - indication de progression



Contre Exemple:



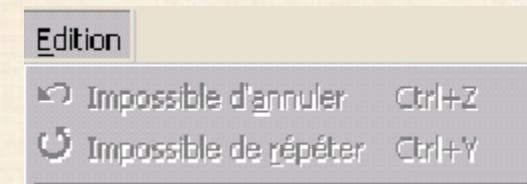
### 3. Critères ergonomiques (Cont. Scapin et Bastien- **Gestion des erreurs**)

**e.3) Correction des erreurs :** moyens mis à la disposition des utilisateurs pour leur permettre de corriger leurs erreurs.

Exemple:



Contre Exemple:



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Homogénéité/cohérence)

#### f) Homogénéité/cohérence

Représente la façon dont les choix de conception de l'interface sont conservés pour des contextes identiques. Le manque d'homogénéité peut augmenter considérablement le temps de recherche. Ainsi cohérence doit être réalisée pour l'apparence, la position, le comportement des différents éléments.

**Exemple :** manque d'homogénéité entre les facettes d'une même application:

##### énoncé de l'exercice 1

**Calculer l'aire d'un rectangle de 7 mètres de largeur et 3 mètres de longueur.**

##### réponse

##### *Exercice 2 : énoncé :*

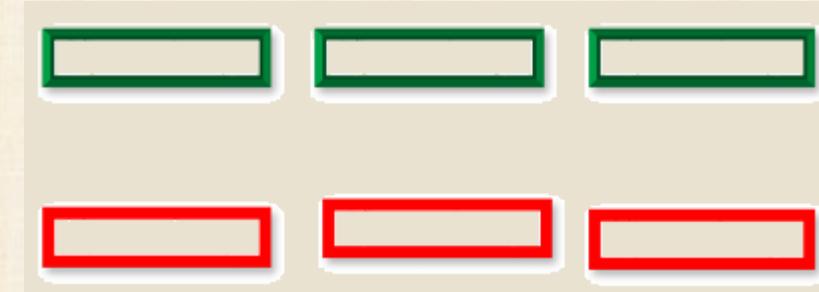
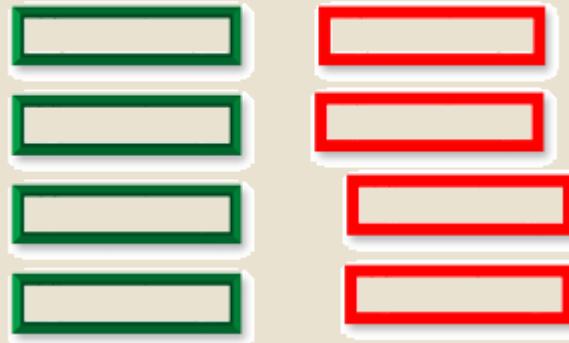
*Calculer l'aire d'un carré de 7 mètres de côté.*

##### *Réponse à l'exercice 2 :*

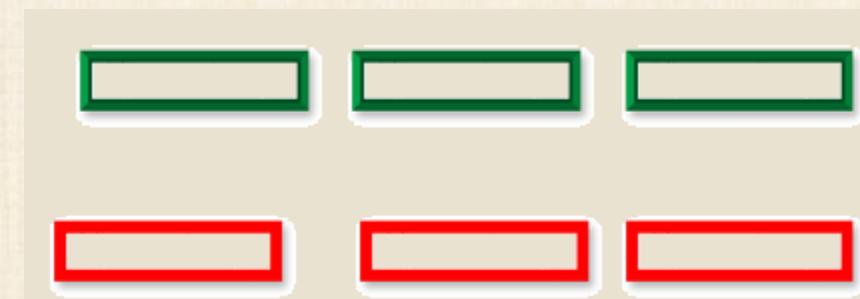
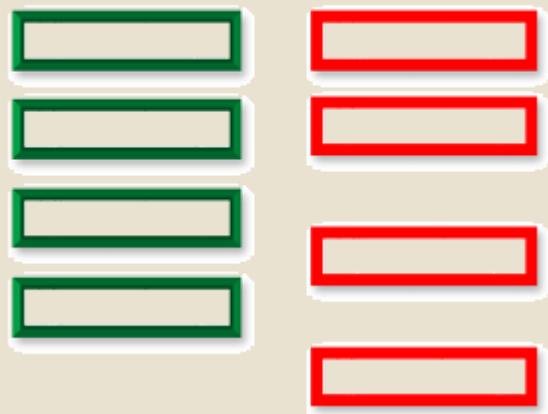
### 3. Critères ergonomiques (Cont. Scapin et Bastien - Homogénéité/cohérence)

Exemple: les éléments d'une même interface doivent être homogènes

- Alignements

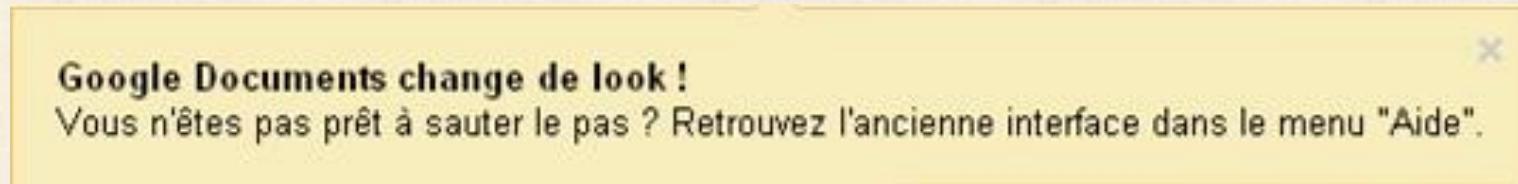


- Espacements

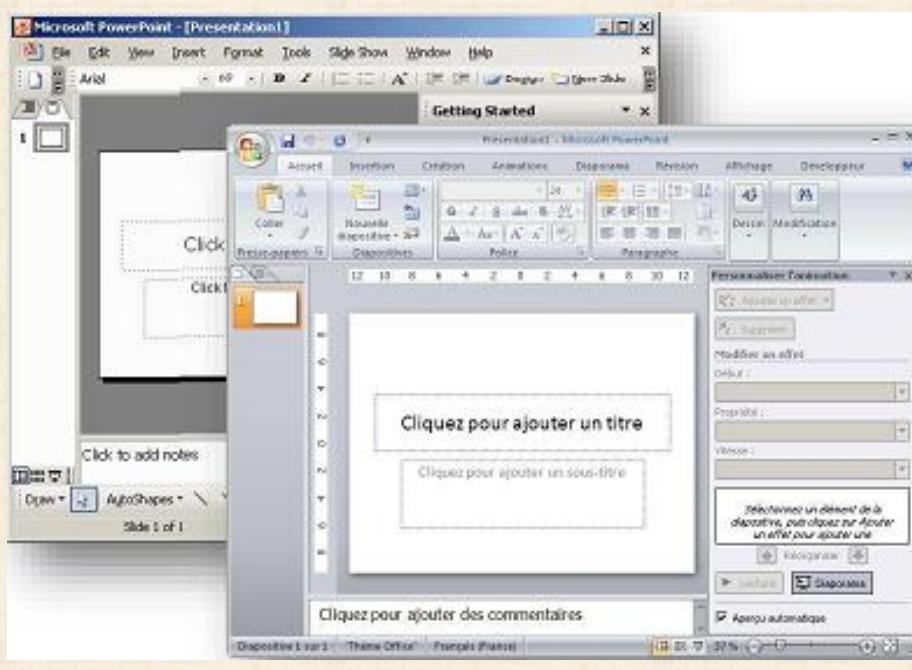


### 3. Critères ergonomiques (Cont. Scapin et Bastien- Homogénéité/cohérence)

Exemple: le respect de la cohérence dans les différentes versions d'une même application



Contre Exemple:



La touche Menu a été remplacée par la touche Applications récentes pour vous permettre d'accéder aux applications récemment utilisées. Appuyez sur Aide pour obtenir plus d'informations.

Ne plus afficher

Aide

OK

Déplacement de commandes entre les versions de MS-Office exp: Office 2003 &2007

### 3. Critères ergonomiques (Cont. Scapin et Bastien- Significance des codes )

**g) Significance des codes et dénominations:** adéquation entre l'objet ou l'information affichée ou entrée et son référent. Lorsque le codage est signifiant, le rappel et la reconnaissance sont meilleurs

Exemple:



Contre Exemple:



Erreur d'application  
affichée comme une erreur  
système.

### 3. Critères ergonomiques (Cont. Scapin et Bastien- Compatibilité)

#### h) Compatibilité

L'accord entre les caractéristiques des utilisateurs (mémoire, perceptions, habitudes, compétences, âge, attentes, etc.) et des tâches, d'une part, et l'organisation des sorties, des entrées et du dialogue d'autre part. La compatibilité concerne également le degré de similitude entre divers environnements ou applications. Les performances sont meilleures lorsque l'information est présentée sous une forme directement utilisable.

**Compatibilité utilisateur-tâche:** à prendre en compte:

- âge
- niveau « technologique »
- handicaps
- fréquence d'utilisation



### 3. Critères ergonomiques (Cont. Scapin et Bastien- Compatibilité)

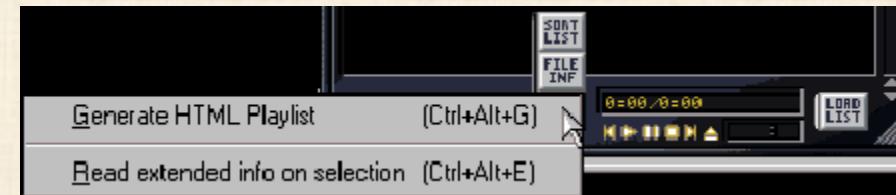
Compatibilité inter-application:

Exemple:



**"We've made significant improvements since you bought your last computer. We switched from beige plastic to dark gray."**

Contre Exemple:



### 3. Critères ergonomiques (Autres critères d'évaluation)

D'autres critères ont été proposés dans différentes approches pour l'évaluation ergonomique des interfaces telles:

- **Meinadier** proposée pour vérifier les performances du couple ordinateur-utilisateur, mesurant la performance en deux cas : Bonne IHM (se caractérise par la facilité d'apprentissage et d'utilisation, le plaisir d'utilisation, acceptabilité du logiciel .etc) et Mauvaise IHM (elle-même se divise en deux niveaux : le niveau individuel (Confusion, panique, stress etc) et le niveau entreprise (Rejet, erreurs graves, mauvaise ou sous-utilisation, régression vers des tâches d'exécution .etc)
- **Shneiderman** comportant 5 critères centraux d'évaluation (comme le temps d'apprentissage, la vitesse d'exécution des tâches le taux d'erreurs .etc),
- **Nielsen** basée sur les heuristiques (10 règles, telles, proposer des dialogues simples, naturels et concis, minimiser la charge mémoire de l'utilisateur, permettre le feedback, rendre accessible aide et documentation etc.)
- et finalement l'approche de **Coutaz** basant sur les sept règles d'or: réduire la charge cognitive, lutter pour la cohérence et la concision, mettre le contrôle entre les mains de l'user, souplesse d'utilisation, structurer le dialogue et prédire les erreurs