

Ergonomie du logiciel

Interfaces Humains-Machine (IHM)

- Ergonomie (du grec ergos, le travail et de Nomos, la loi): discipline qui vise l'adaptation d'un système à son utilisateur, afin que ce dernier puisse mener ses activités avec un maximum d'efficacité, de satisfaction et de bien-être, avec une phase d'adaptation réduite.

Référence bibliographique

AFNOR (2003). *Ergonomie de l'informatique. Aspects logiciels, matériels et environnementaux*, Recueil Normes Informatique, ISBN 2-12-236211-1.

Qu'est-ce que l'ergonomie ?

- Exemple de la mise en page d'une bible (d'après Alain GIBOIN)

Question : **pourquoi cette mise en page ?**



La réponse est dans la situation de lecture

Bible posée sur un lutrin

- lecture debout...
- fatigue et besoin de trouver un appui...
- l'appui : la bible...



Lecture debout, les mains appuyées sur la bible



Les marges sont là pour permettre de s'appuyer sans gêner la lecture

Qu'est-ce que l'ergonomie ?

■ Exemple de la mise en page d'une bible

Rappel de la question : **pourquoi cette mise en page ?**



■ Réponse : **pour adapter l'objet à son utilisateur**

Adaptation à l'utilisateur = Ergonomie

Qu'est-ce que l'ergonomie ?

**Ergonomie =
adaptation d'un
objet/outil à son
utilisateur**

Logiciels de
grossissement de
caractères pour
personnes
handicapées
visuelles



Interfaces Humains-Machine (IHM)

- Pourquoi faut-il s'intéresser à la conception et au développement des Interfaces Humains-Machine (IHM)?
 - Une mauvaise interface peut provoquer le rejet de la part des utilisateurs, leur frustration, voire leur anxiété, face au système qu'ils ont à utiliser.
 - Inversement, une bonne IHM amplifie les sensations positives de succès et de contrôle.
- Mais qu'est-ce qu'une bonne IHM ?
 - D'une façon générale, une bonne IHM est une interface que l'utilisateur ne remarque plus

Objectifs de la conception des IHM (1)

- Concrètement, les objectifs généraux pour la conception d'une IHM sont :

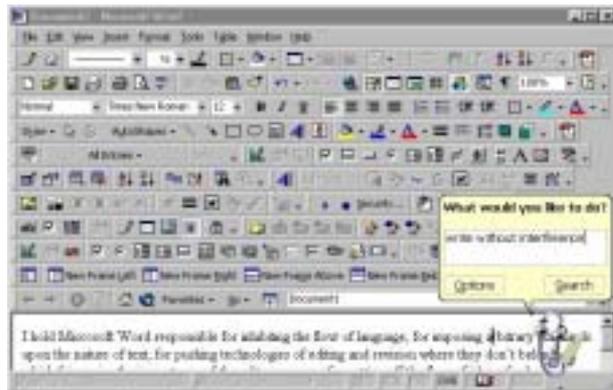
(US Military Standards for Human Engineering Design Criteria)

- permettre la réalisation des tâches prévues,
- minimiser l'investissement nécessaire pour pouvoir utiliser l'interface
- garantir des interactions fiables
- favoriser la standardisation

Objectifs de la conception des IHM (2)

- Permettre la réalisation des tâches prévues:

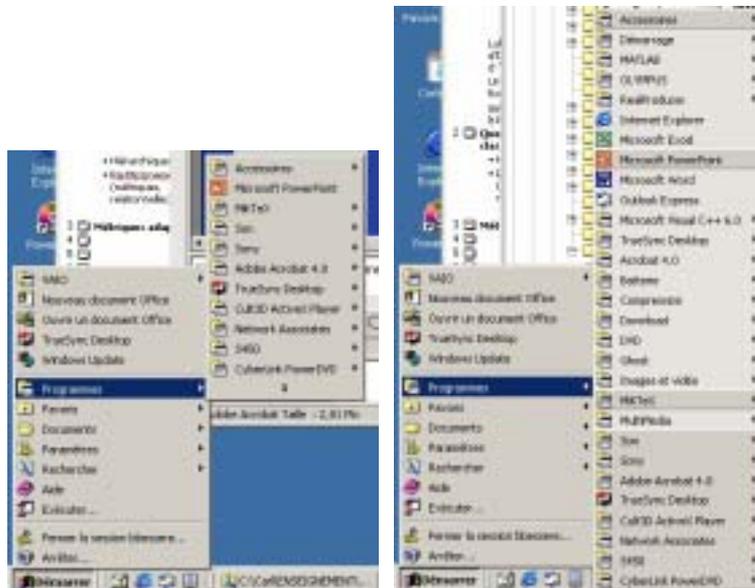
- La richesse fonctionnelle doit être adaptée.
Trop peu de fonctions → système inutilisable
Richesse fonctionnelle trop importante → système difficile à maîtriser.



Analyse fonctionnelle

- Une analyse fonctionnelle doit être réalisée pour recenser l'ensemble des tâches et sous-tâches véritablement nécessaires, ainsi que leur fréquence d'utilisation.
- "Analyse fonctionnelle dynamique"
Règle du MRU (Most Recently Used) (exemple Win2k, pour les programmes les plus utilisés)

MRU : Exemple Win2K



Objectifs de la conception des IHM (3)

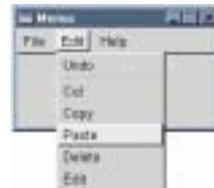
- Minimiser l'investissement de l'utilisateur:
 - Minimiser la durée d'apprentissage et le niveau de compétences requis, ce qui peut s'obtenir, entre autre, par un mimétisme plus ou moins marqué avec d'autres IHM auxquelles l'utilisateur à déjà été confronté

Exemple :
le drag & drop
(glisser & déposer)
à la souris

ou au stylet

Objectifs de la conception des IHM (4)

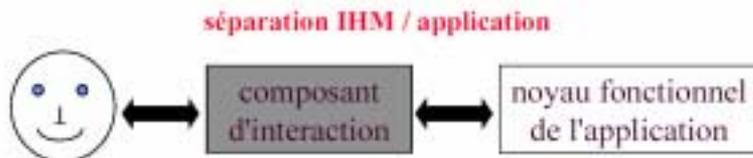
- Garantir des interactions fiables :
 - Garantir un bon degré de fiabilité lors des interactions.
 - La confiance que place l'utilisateur dans le système est souvent fragile! Les interactions offertes par une bonne IHM doivent donc contribuer à augmenter la confiance de l'utilisateur : fonctionnement sans erreurs, organisation fonctionnelle claire et cohérente, stabilité dans le temps, ...
- Favoriser la standardisation :
 - La standardisation permet de réduire les temps d'apprentissage, augmente la confiance et les performances des utilisateurs (moins d'erreurs) et améliore la portabilité des systèmes.
 - Tcl/Tk, GTK, Llama, ect...



Méthodes de conception

■ Règles :

- Séparer la conception de l'application de la conception de l'interface
- Prendre en compte les utilisateurs
- Concevoir de manière itérative, avec phases d'affinement progressif
- Par une équipe pluridisciplinaire



Modèle de Seeheim

Domaines connexes

■ Le développement d'IHM est une activité multidisciplinaire. Il faut des spécialistes:

- **en psychologie et facteurs humains**
pour prendre en compte les concepts issus des théories de la perception et de la cognition;
- **en conception logicielle**
pour utiliser au mieux les techniques informatiques disponibles;
- **en développement de matériel**
pour mettre à profit les progrès dans le domaine de la conception de nouveau périphériques, et offrir un accès au système au plus grand nombre de personnes (handicapés compris)
- **en conception graphique**
pour la fabrication des «layouts» qui seront utilisés dans le cas d'interfaces visuelles
- en traitement de la parole, en ergonomie, ...

Principales étapes de la conception

- Les principales étapes de la conception d'une IHM sont:
 - 1) déterminer l'ensemble des tâches que l'IHM devra permettre de réaliser :
une bonne IHM est une IHM dont les objectifs fonctionnels sont clairement identifiés;
 - 2) déterminer les caractéristiques principales des utilisateurs qui seront amenés à utiliser l'IHM (leur profil) :
la qualité d'une IHM est directement dépendante de son adéquation avec la population d'utilisateurs pour laquelle elle est prévue;
 - 3) proposer plusieurs prototypes d'interface qui seront discutés et évalués par les concepteurs et les utilisateurs potentiels :
une bonne IHM naît le plus souvent de la diversité... et plusieurs pistes doivent donc être explorées;

Principales étapes de la conception

- 4) produire une spécification explicite de l'IHM, décrivant à la fois les contraintes fonctionnelles et les contraintes de layout; un manuel d'utilisation et une référence technique pourront également être produits durant cette phase;
- 5) réaliser l'IHM proprement dite (phase d'implémentation effective);
- 6) évaluer l'IHM produite sur la base d'indicateurs reconnus

Loi de Fitts

- Loi de Fitts
- En ergonomie, la loi de Fitts est un modèle du mouvement humain.
- Elle prédit le temps mis pour se déplacer rapidement d'un point à une zone cible, en fonction de la distance de la cible et de sa taille
- Utiliser pour modéliser l'acte de pointer
 - Monde réel
 - Sur les IHM
- Publié par [Paul Fitts](#) en [1954](#).



Tâche de pointage (à gauche) et de traversée de but (à droite).

Loi de Fitts

- Le modèle (formulation de Shannon) pour un mouvement dans une direction

$$T = a + b \log_2 \left(\frac{D}{W} + 1 \right)$$

avec

- * T temps moyen pour effectuer le mouvement
- * a et b sont des constantes empiriques
- * D: distance du point de départ au centre de la cible
- * W largeur de la cible dans la direction du mouvement (tolérance)

⇒ Compromis vitesse/précision

Loi de Fitts

Exercice

Voici des temps en secondes mesurés lors d'une expérimentation avec un dispositif particulier. Déterminer (grossièrement) les constantes de la loi de Fitts

$$T = a + b \log_2 \left(\frac{D}{W} + 1 \right)$$

d \ D	1.00	5.00	10.00
0.10	0.31	0.45	0.56
0.50	0.13	0.30	0.38
1.00	0.07	0.22	0.32

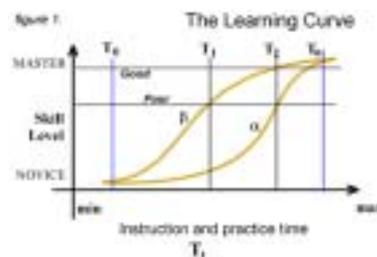
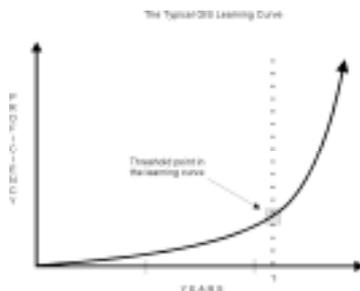
D	1	1	1	5	5	5	10	10	10
d	0,1	0,5	1	0,1	0,5	1	0,1	0,5	1
2D/d	20	4	2	100	20	10	200	40	20
log2(2D/d)	4,32	2	1	6,64	4,32	3,32	7,64	5,32	4,32

Critères d'évaluation

Dans le but de quantifier (mesurer) la qualité d'une IHM, quelques indicateurs sont fréquemment employés:

- **La durée d'apprentissage :**

la durée moyenne nécessaire pour qu'un utilisateur typique maîtrise les fonctions pour lesquelles l'IHM a été développée;



Critères d'évaluation

- **La rapidité d'exécution :**
la durée moyenne de réalisation d'un ensemble test de tâches par un groupe d'utilisateurs de référence;
- **Le taux d'erreur :**
le nombre et la nature des erreurs faites par le groupe d'utilisateurs de référence lors de la réalisation de l'ensemble test de tâches ;
- **La mémorisation dans le temps :**
c'est-à-dire l'évolution dans le temps des critères précédents (qu'en est-il après une heure d'utilisation, un jour, une semaine, ... ?);
- **La satisfaction subjective :**
la satisfaction subjective des utilisateurs pourra être mesurée par le biais de questionnaires ou d'interviews face-à-face.

Critères d'évaluation

- Il n'est souvent pas possible d'optimiser l'ensemble des critères mentionnés ci-dessus et des compromis devront être faits selon les domaines d'application des IHM
 - Outils automatiques d'évaluation
 - Monkey Test
- Test d'utilisabilité (suivant une norme: ex. ISO 9241-11)
- On considère qu'une IHM est utilisable lorsque :
- l'utilisateur peut réaliser sa tâche (efficacité)
 - qu'il consomme un minimum de ressources pour la faire (efficience)
 - et que l'interface est agréable à utiliser (satisfaction)

Évaluation heuristique

- **Promenade (*walkthrough*) informelle**
 - A partir d'une liste de principes (*heuristics*)
 - inspecter l'interface
 - afin d'identifier des problèmes potentiels d'utilisabilité

Évaluation heuristique

- **Promenade (*walkthrough*) informelle**
 - A partir d'une liste de principes (*heuristics*)
 - inspecter l'interface
 - afin d'identifier des problèmes potentiels d'utilisabilité

Évaluation heuristique

■ Avantages et inconvénients

- Economique et facile à appliquer (+)
 - Aucun utilisateur n'est requis
 - Peut être réalisée individuellement
 - Nécessite une planification minimale
- Utilisable très tôt dans le cycle de conception (+)
- Problèmes liés au domaine de l'application difficilement identifiables (-)

« *E.H. = a discount usability engineering method* »

Évaluation heuristique

■ Méthode

- Un groupe d'évaluateurs (plus on est, plus on trouve des problèmes)
- 3-5 personnes
- Des « heuristiques » de conception facilitent l'inspection

Évaluation heuristique

■ Heuristiques initiales

- Dialogue simple et naturel
- Parler le langage des utilisateurs
- Ne pas surcharger la mémoire des utilisateurs
- Feed-back
- Sorties clairement indiquées
- Raccourcis
- Messages d'erreur appropriés
- Prévenir les erreurs
- Aide et documentation

Évaluation heuristique

■ Dialogue simple et naturel

- La manière d'interagir avec le système doit être évidente
- Regrouper les objets de l'interface
- Le nombre d'informations à l'écran doit être suffisant
- Vue générale et détails
- Éviter les « décorations » qui perturbent

Évaluation heuristique

■ Parler le langage des utilisateurs

- Utiliser le langage du domaine
- Éviter le langage informatique
- Utiliser des métaphores familières
- Utiliser des icônes, images, etc., en rapport avec le domaine

Évaluation heuristique

■ Ne pas surcharger la mémoire des utilisateurs

- Fournir suffisamment d'information à l'écran
- Privilégier la reconnaissance plutôt que le rappel
- Mettre en valeur l'information importante
- Fournir des valeurs par défaut

Évaluation heuristique

■ Cohérence

- Cohérence de la disposition des informations a l'écran (*layout*)
- Cohérence de l'interaction
- Cohérence du langage
- Cohérence des fonctionnalités

Évaluation heuristique

■ Feed-back

- Indiquer quand le système est en train de faire un traitement
- Fournir un feed-back aux actions de l'utilisateur
- Fournir un feed-back en rapport avec la tâche de l'utilisateur
- Indiquer quand le système a échoué

Évaluation heuristique

■ Sorties clairement indiquées

- Montrer le chemin suivi par l'utilisateur dans le système
- Montrer comment aller en avant et en arrière
- Montrer comment s'orienter dans l'espace des informations

Évaluation heuristique

■ Raccourcis

- Utiliser des raccourcis pour réaliser les actions
- Inférer la suite d'une frappe clavier
- Autoriser les sauts vers les endroits où l'utilisateur veut aller
- Réutiliser l'historique de l'interaction

Évaluation heuristique

■ Messages d'erreurs pertinents

- Éviter d'utiliser un langage codé

Exemple : « Erreur No 31 »

- Utiliser le langage des utilisateurs

Évaluation heuristique

■ Prévenir les erreurs

- Engager un dialogue système-utilisateur lorsque des actions de l'utilisateur peuvent conduire à des conséquences désastreuses
- Indiquer le statut du système et de ses actions
- Utiliser les commandes « *undo* » et « *redo* »

Évaluation heuristique

■ Aide et documentation

- Les manuels papier ne devraient pas être nécessaires
- Fournir l'aide en ligne à la demande de l'utilisateur
- Sinon, fournir l'aide à l'initiative du système

Évaluation heuristique

■ Estimation de la gravité des problèmes

- 0 = Ceci n'est pas un problème d'utilisabilité
- 1 = Problème « cosmétique » uniquement : pas nécessaire de le régler si l'on n'a pas le temps
- 2 = Problème mineur : problème à régler avec une priorité faible
- 3 = Problème majeur : important à régler (priorité élevée)
- 4 = Problème « catastrophique » : à régler impérativement avant de diffuser l'interface

Approche heuristique: exemple

- Etude de cas par approche heuristique: exemple d'évaluation:

http://www.otierney.net/hci2/heuristic_evaluations.html

Domaines d'application (1)

- Les principaux domaines d'application pour les IHM se répartissent dans trois grandes catégories:
 - les systèmes critiques (généralement temps réel)
 - les systèmes commerciaux et industriels
 - les systèmes personnels

Domaines d'application (2)

■ Les systèmes critiques:

(contrôle aérien, pilotage d'avions ou de centrales nucléaires, appareillage médical, ...)

- Dans ce domaine, la fiabilité (faible taux d'erreur) et la performance (temps de réponse très court) sont centrales, y compris et en particulier dans des conditions de stress pour les utilisateurs.
- Elles seront souvent obtenues au prix de durées d'apprentissage plus longues, et la mémorisation est garantie par des entraînements fréquents



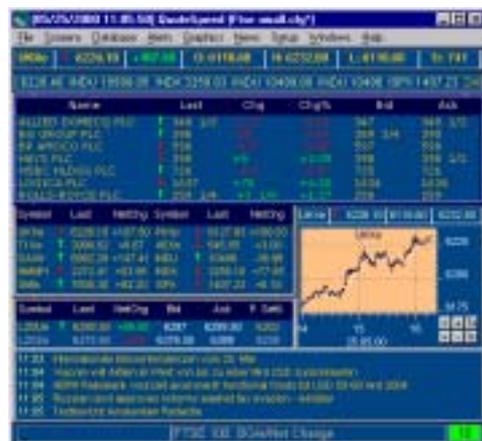
Domaines d'application (3)

■ Les systèmes commerciaux et industriels: (applications de type transactionnels, dans le domaine des banques, assurances, gestion des stocks, comptabilité, réservations vente, ...)

- facteur déterminant : le coût.

La formation des utilisateurs coûte cher; la durée d'apprentissage devra donc être réduite... tout en garantissant cependant des taux d'erreurs acceptables (car les erreurs représentent également

- un levain de l'exécution est également importante (nombre d'opérations réalisées.)



Domaines d'application (4)

- Les systèmes personnels et de loisirs:
(suites bureautiques, jeux, applications éducatives, systèmes d'exploitation, ...)
 - La **satisfaction subjective** des utilisateurs est l'élément central car c'est souvent le critère déterminant pour le choix d'un système... et la concurrence est féroce.
 - La **rapidité de l'apprentissage** et de faibles **taux d'erreurs** seront également déterminants.
 - Difficulté : **l'extrême variété des utilisateurs** (du néophyte à l'expert confirmé), qui va souvent nécessiter une structure à plusieurs niveaux pour les interfaces (*multilayered interface*)



Exemple : multilayered interface



Exemple : *MAC OS 10*

- Zone de meilleure accessibilité/ visibilité

