

Brève excursion dans les IPM

Ou que faut-il retenir de cette tâche
complexe et quasi impossible:
concevoir une bonne IPM

Contenu de la présentation

- *Software Design Manifesto*
- Heuristiques
- Modèles, encore !
- Facilité d'utilisation

Conception

- *Bringing Design to Software*, sous la direction de Terry Winograd, édité par Addison-Wesley
- Chapitre : *Software Design Manifesto*
- « Conception »
 - Acception de ISO 12207 versus celle du *Manifesto*

Manifesto : quelques points (1/2)

- De gros problèmes continuent à exister par rapport aux IPM (Et ce ne sont sûrement pas les informaticiens qui savent les voir : ils minimisent toujours toutes les difficultés)
- Les compagnies essaient surtout de stabiliser les produits.
- L'utilisateur n'ose pas se révolter et pense que c'est sa faute si l'ordinateur ne fonctionne pas bien
- Il faut reconnaître le rôle central du concepteur de l'IPM.
- Jusqu'à présente ce sont surtout les informaticiens qui ont conçu les IPM (comme si les ingénieurs concevaient l'esthétique des autos... mais les informaticiens ne sont pas formés pour cela...)

Manifesto : problèmes (2/2)

- Il faut repenser les fondements de la réalisation du logiciel.
- Le *Designer du logiciel* (trouver un autre nom ?) doit être celui qui a la pleine responsabilité du produit.
- Dans les entreprises il faut ouvrir des carrières pour ce genre de personnes. Ça doit être une profession à part entière.
- Il faut développer des outils pour le concepteur (ici c'est la place de l'ingénieur du logiciel !).
- Le concepteur doit avoir une formation technique pour pouvoir dialoguer avec les informaticiens.

Heuristiques

Heuristiques : 1) s'adapter aux types d'utilisateurs et à leurs connaissances

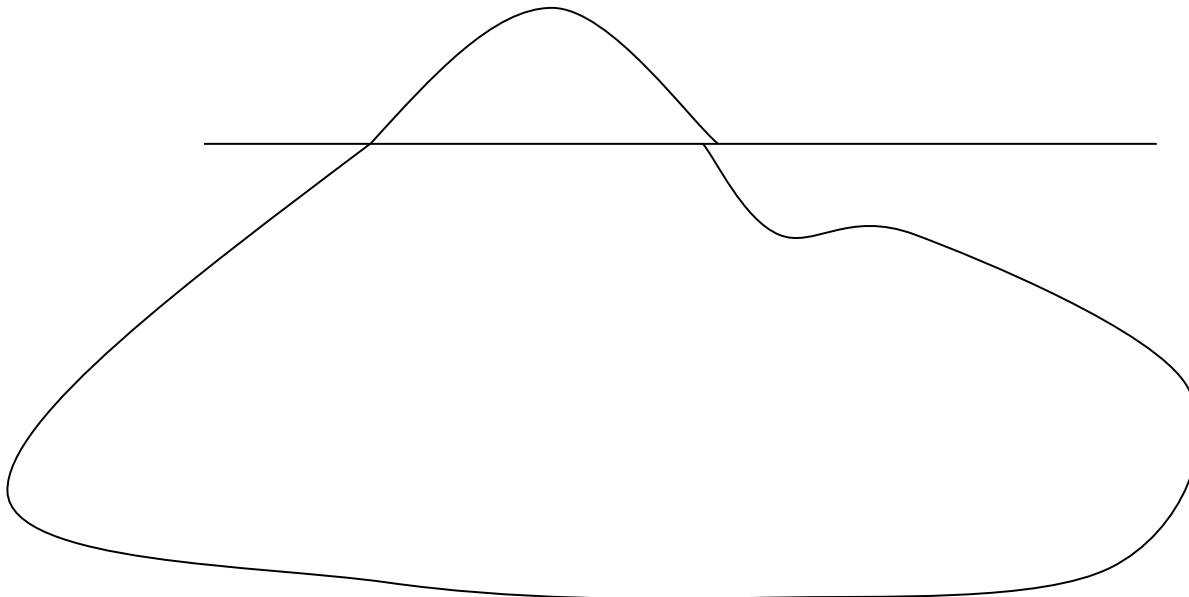
- Caractéristiques
 - Niveau de scolarisation,
 - Le type de travaux qu'ils exécutent,
 - La fréquence d'emploi du système,
 - Expertise, etc.
- Connaissances du concepteur
 - Quelques connaissances de psychologie
- Impossibilité ?
 - Concevoir une IPM pour monsieur tout le monde
- Danger
 - Penser que l'utilisateur est comme nous (les ingénieurs du log.)
- Exemple :
 - Touches de raccourci pour des utilisateurs experts

Heuristiques : 2) Cohérence

- Facilite le raisonnement par analogie
- Le même type d'action doit toujours donner le même type de résultat.
- **Interne** :
 - Exemple de manque de cohérence « utile »
- **Externe**
 - compatibilité entre les produits.
 - Normes

Heuristiques : 3) Adaptation à la tâche

- Objets qui font partie de son monde
- Pas des objets qui ont été créés par les informaticiens et qui sont censés l'aider.
- « L'Iceberg d'IBM »



Heuristiques : 3) ni première ni dernière

- Tenir compte du « passé »
- Considérer les possibilités technologiques
- Un jour elle sera critiquée
 - C'est la vie!

Heuristiques : 4) Changement de tâche

- Faciliter le changement de tâche
- L'IPM ne doit pas être un frein à la progression « normale »
- Exemple de Windows

Heuristiques : 5) règle de Miller

- ENCORE !!!!!!!!!!!
- Max sept ou huit éléments par fenêtre
- Grouper les items (cadres)
- Pas de listes trop longues

Heuristiques : 6) utilisateur en contrôle de l'application

- L'humain doit sentir qu'il contrôle la machine
 - Cela peut compliquer l'interface
- Importance des événements
- Note pour nous les programmeurs
 - Programmation événementielle.

Heuristiques : 7) **possibilité d'expérimenter**

- Apprendre en travaillant
- Possibilité de défaire (undo)... sans fin
- Messages de confirmations très clairs.

Heuristiques : 8) **aide contextuelle**

- Favoriser l'**apprentissage** en travaillant,
- L'aide si bien intégrée que l'utilisateur ne sait pas s'il est dans " l'aide " ou dans la " vraie " application.
- Lorsque l'utilisateur est dans l'aide il doit pouvoir saisir des données pour l'application

Heuristiques : 9) **sablier**

- Ne jamais laisser l'utilisateur sans des réponses pour plus que 1..3 secondes
- Sablier (ou équivalent) comme signe d'attente.
- La lumière d'accès au disque dur ne suffit pas !

Heuristiques : 10) **Jamais d'erreurs de l'utilisateur**

- L'utilisateur ne se **trompe jamais**.
- D'une manière moins radicale : la machine doit être très **robuste** et ne pas faire payer trop cher les erreurs.
- Il faut être très "gentil" dans la gestion des soi-disant erreurs
- Si une catastrophe arrive, elle soit bien annoncée !

Heuristiques : 11) normes

- Il faut que toute entreprise travaillant dans le domaine des IPM crée des **normes**
- Normes à intégrer aux normes des constructeurs
- À leur tour à intégrer dans les normes internationales.
- Chaque individu aura aussi ses “normes” personnelles (pratiquement son style) qui doivent s’intégrer avec celles de l’entreprise

Air France

1. Info pilote versus info passagers
2. 152 versus 50

Modèles Encore !

Utilisateur

- Trois axes
 - Connaissances du domaine
 - Connaissances du système (machine)
 - Connaissance des ordinateurs
- Trouver sa place dans l'espace
expérience/connaissance

Mais... qu'est-ce qu'un humain (Modèle informatique)

Ordinateur

- Registres
- Mémoire principale
- Mémoire de masse en ligne
- Mémoire de masse hors ligne
- Clavier
- Dispositifs de reconnais. de la voix
- Écran tactile
- ?
- Écran, imprimante
- Synthétiseur de la voix
- Système d'exploitation
- Application
 - Et pourquoi pas (sic!)
- Environnement

Humain

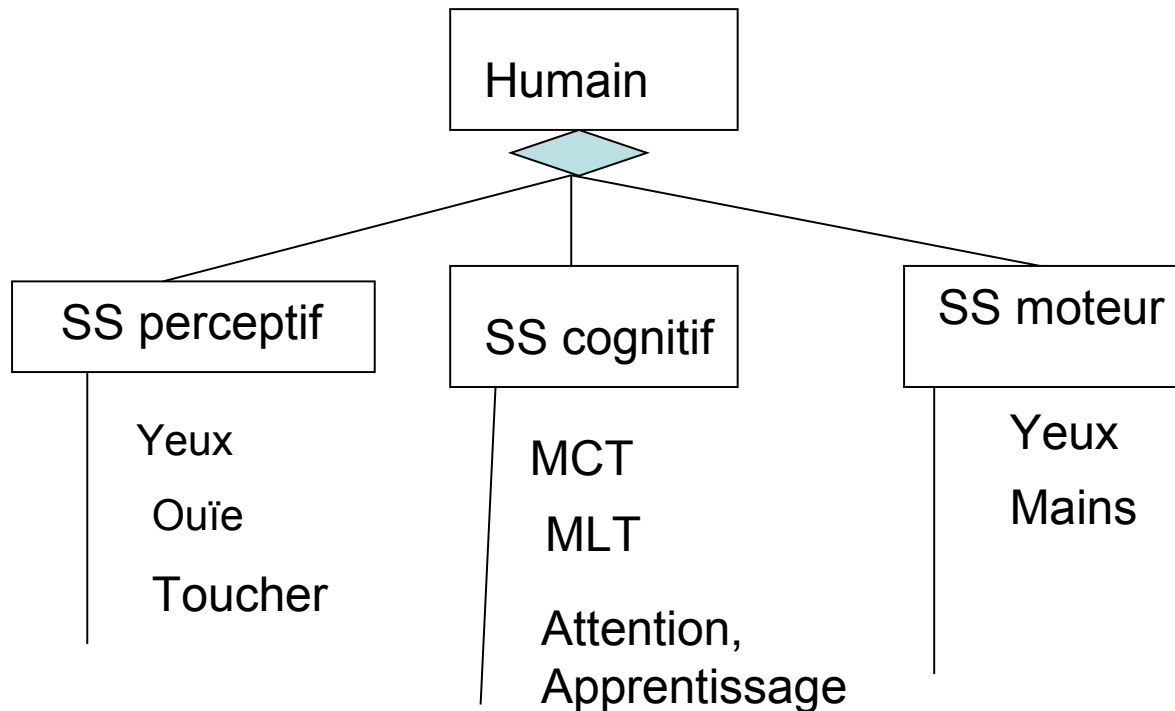
Mémoire à court terme
Mémoire intermédiaire
Mémoire à long terme
Livres, journaux...
Yeux
Oreilles
Toucher
Nez, goût
Mains, expressions du visage
Voix
Unité de contrôle
Connaissance et savoir-faire

Société

Modèle informatique (réflexion)

- Avant
 - L'ordinateur comme un humain
- Maintenant
 - L'humain comme un ordinateur
- Drôle de façon de procéder !

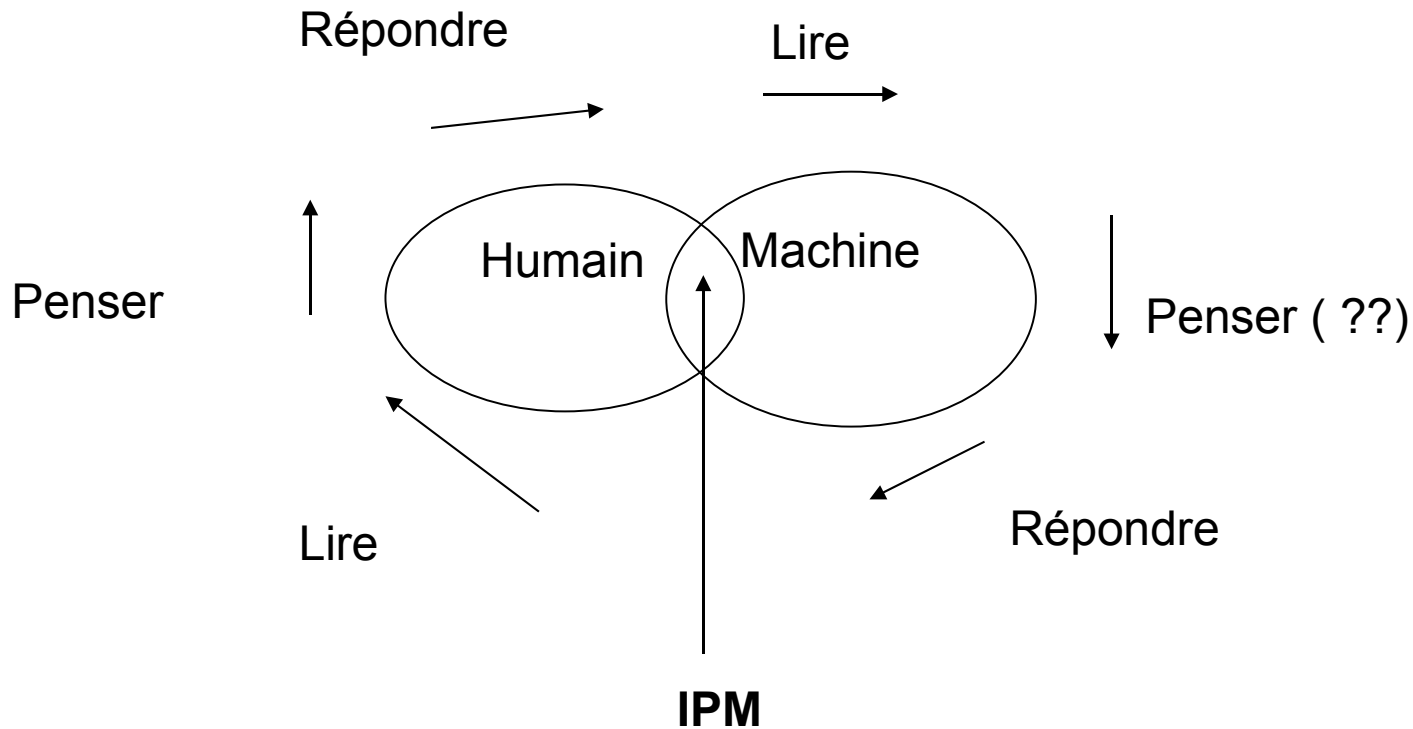
Humain presque'en UML



Utilité du modèle informatique

- Ressources
- Limitation ressources
- IPM afin que l'humain consomme le moins de ressources possibles
- Pour les deux genres de tâches
 - Fonctionnelles : « transformer » le domaine comme si la machine n'existait pas
 - Opérationnelles : ce qu'il faut faire avec la machine pour atteindre le domaine

Lire, penser (?), répondre



MCT (1/2)

- Perd facilement l'information (surtout si on est distrait par d'autres informations ou activités);
- Plus l'activité distrayante est complexe et plus on oublie;
- Plus on vieillit et moins d'informations on peut stocker;
- C'est plus facile de se rappeler des concepts quand ils sont présentés sous forme de images que comme suite de lettres

MCT (2/2)

- Diviser en groupe et hiérarchies avec un maximum de sept ou huit éléments.
- Des mots qui riment sont plus difficiles à différencier
- Les étiquettes et les noms des objets doivent être très différents
- le nombre interaction doit être réduit au minimum quand on doit retenir de l'information dans la MCT.
- Tenir compte de l'âge. Après 60 ans, en moyenne, la capacité de la MCT est beaucoup plus faible.
- Ajouter, quand il est possible, des images pour faciliter le rappel

MLT (1/2)

- Sa capacité est à peu près illimitée comme la durée de stockage (limite la mort).
- Les informations sont stockées de manières très complexes (et assez inconnue)
- Les mécanismes de récupération sont lents, non fiables (exemple de films) et difficiles.
- Il est plus facile de récupérer les informations que l'on emploie souvent

MLT (2/2)

- Faire apprendre l'utilisateur en le faisant agir (exemples).
- Structurer l'information de manière que l'utilisateur puisse en induire facilement les buts et sa position dans le " chemin " vers la solution du problème.
- Bien séparer ce qui est différent (couleurs, formes, etc.).
- Emploi de mots «concrets» liés à sa tâche plutôt qu'abstraites.

Processus

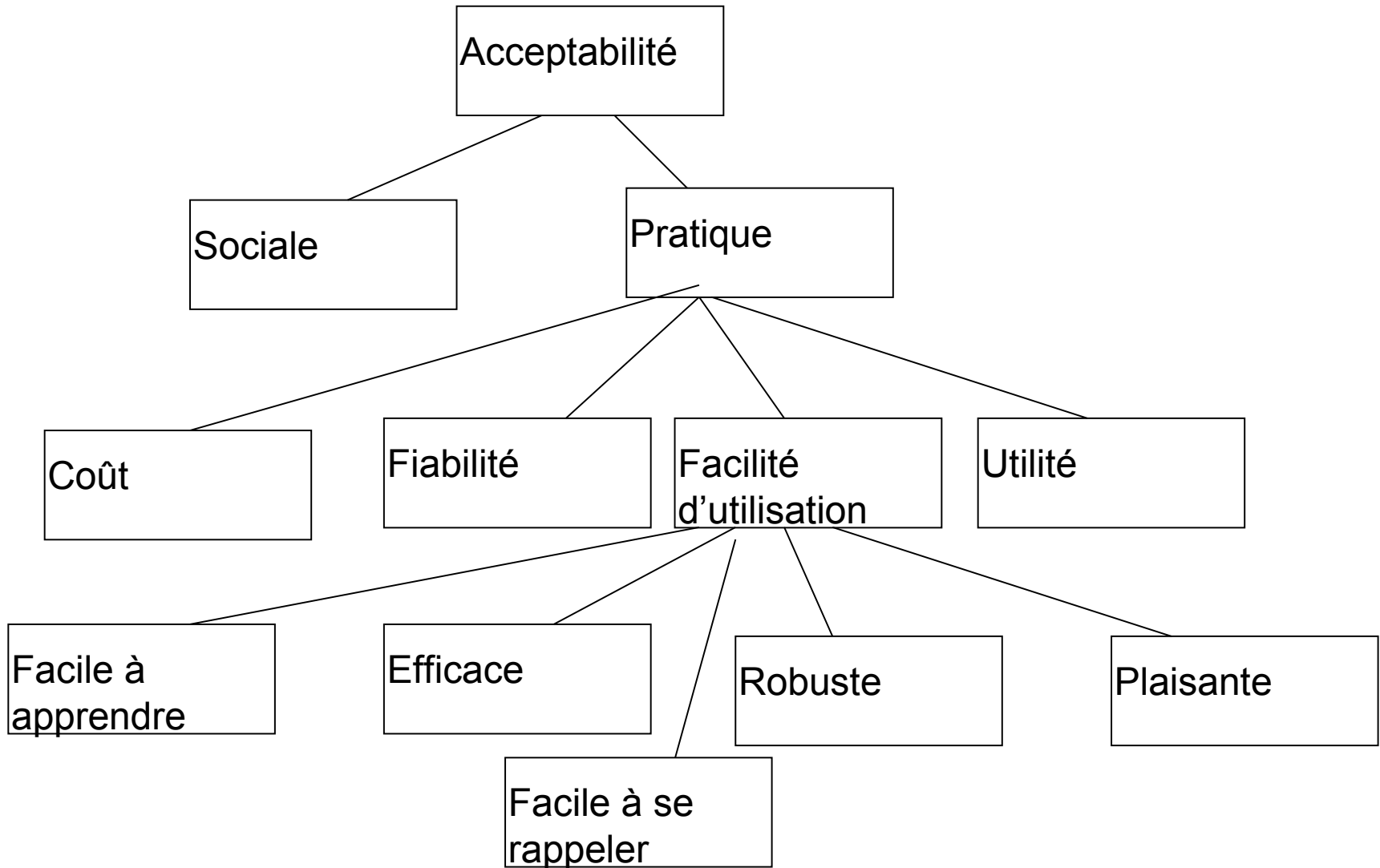
- Attention sélective
 - Clignotement (arbres de Noël)
 - Habitude
- Apprentissage
 - Structuration
 - Analogie
- Solution de problèmes
 - Objectifs, stratégie
 - Peu d'algorithmes conscients

Facilité d'utilisation

Ergonomie

- « **L'ergonomie** se propose de concevoir le travail humain de manière qu'il se situe à l'intérieur de la marge d'adaptabilité de l'homme. Cette marge étant définie en fonction de critères physiologiques, psychologiques et socio-économiques, l'ergonomie est tributaire de diverses disciplines scientifiques. À la suite de l'évolution des techniques industrielles, le domaine de l'ergonomie s'est considérablement élargi. Limitée, à l'origine, à l'étude des problèmes énergétiques de l'homme au travail, l'ergonomie a considéré, par la suite, l'influence des contraintes psychiques, le rôle de l'environnement et le jeu des systèmes relationnels. » J. MARCELIN, in La Recherche, no 4, sept. 1970, p. 368

Facilité d'utilisation



Cycle de vie de la facilité d'utilisation (1/2)

1. Comprendre la toile de fond du travail à automatiser
 - Lecture des documents du projet, rencontre avec les gestionnaires et les représentants du client
 - Identifier les acteurs, leurs requêtes et les réponses de la machine
2. Analyse de la tâche
 - Étudier les différentes activités
 - Minimum de perturbation
 - Stocker les informations
3. Étude des utilisateurs
 - Entrevues

Cycle de vie de la facilité d'utilisation (2/2)

1. Étude des produits
 - Jusqu'à « japoniser »
2. Fixation des objectifs de la facilité d'utilisation
3. Conception
4. Proto
5. Évaluation de l'IPM
6. (Cycler à partir de 6)
7. Suivi