

## IHM – DI Polytech'Tours



J.Y. RAMEL

LI - RFAI

## Bibliographie

- Coutaz J. Interfaces Homme-Ordinateur, Dunod.1990.
- Clayton L. et J. Rieman. «Task-centered User Interface Design : A Practical Introduction». 1994. "ftp://ftp.cs.colorado.edu/pub/distribs/clewis/HCI-Design-Book"
- Évaluation heuristique : <http://www.useit.com/papers/heuristic>
- Kolski C. Interfaces Homme-Machine, Hermès. 1997
- Nanard M. Facteurs humains dans la conception de sites Web. Tutorial IHM 2000. Mprntpellier.
  
- Beldame M. Rapport de DEA. Ecole Centrale de Lyon. 2001.
- Caelen J. Exposé sur la multimodalité. Mai 2000. Grenoble.
- Dufresne A. Projet CINEMA. 2000.  
<http://www.fas.umontreal.ca/com/com3561/bta/indexf2.html>
- Fattersack M. Cours d'IHM. Université Paris V. 1999.
- Prévot P. Cours d'IHM du DEA ISCE 2001. INSA de Lyon
- Tarpin-Bernard F. Cours d'IHM de GPr. 2001. INSA de Lyon.

## Bibliographie

### • Référence citées :

- [Nielsen94] "Usability Inspection Methods« . Jakob Nielsen, John Wiley & Sons, 1994
- [Norman86] "User Centered System Design" par D.A. Norman et S.W. Draper, Lawrence Erlbaum Associates, 1986
- [Rasmussen86] Information Processing and Human-Machine Interaction. Amsterdam, North-Holland, 1986
  - [Card, Moran, Newell. 1983] S. Card, T. P. Moran, and A. Newell. The Psychology of Human-Computer Interaction. Lawrence Erlbaum Associates, Hillsdale (NJ), 1983.
- [Buisine99] Vers une démarche industrielle pour le développement d'Interfaces Homme-Machine : De l'analyse de l'activité à la génération du code Arnaud Buisine, Thèse de doctorat
- [Devarnay99] Conception et évaluation d'IHM.
- [Nigay2000] Dubois, E., Nigay, L., Troccaz, J. Continuité de l'interaction : une nouvelle exigence pour la multimodalité. Acte du colloque sur la Multimodalité, Grenoble, France, May 2000, p. 65-68.
- [Wharton, Rieman, Lewis, Polson, 1990]. The Cognitive Walkthrough Method: A Practitioner's Guide 1994. In J. Nielsen and R. Mack (Eds). Usability Inspection Methods. New York: John Wiley and Sons.
- [Nielsen, Molich, 1990] Heuristic evaluation of user interfaces, Proc. ACM CHI'90 Conf. (Seattle, WA, 1-5 April), 249-256.
- [Sperandio83] Sperandio, J-C., Bouju, F., L'Exploration Visuelle de Données Numériques Présentées sur Ecran Cathodique, Le Travail Humain,1983, 46, 1, 49-63.

## Objectif du cours

- Introduction : Interaction et IHM
- Principes pratiques de conception d'interfaces
  - **Les règles**
  - **Styles d'interaction**
- Production de systèmes interactifs
  - **Modèles et Méthodes**
  - **Architectures**
  - **Stratégie d'implantation**
- Plasticité des Interfaces, multimédias et hypermedias
  - **Adaptabilité et adaptativité**
  - **Interfaces & Hypermedia**
  - **Stratégies médiatiques**
- Facteurs humains et IHM
  - **Psychologie et ergonomie cognitive**
  - **Modèles : Processeur humain, GOMS, keystroke, ...**
- Évaluation d'IHM (si on a le temps)
  - **Sans utilisateur**
  - **Avec utilisateurs**
  - **Le cas « site Web »**
  - **Normes**

## Interaction et IHM

- **L 'interaction homme machine** désigne l 'ensemble des phénomènes physiques et cognitifs qui interviennent dans la réalisation de tâches avec le concours de l 'ordinateur.
- **L 'interface homme machine** désigne un assemblage de composants logiciels et matériels qui permet l 'accomplissement de tâches avec le concours de l 'ordinateur

## Pourquoi les IHM sont importantes ?

- **Le paradoxe de la productivité :**

- Les investissements des entreprises dans les technologies de l 'information sont énormes
- Les gains de productivité diminuent régulièrement

Productivité = Qtté moyenne de produit / Temps moyen de réalisation

D'après [Futtersack98], tiré de "Taking Computers to Task", par W. Wayt Gibbs, Scientific American, Juillet 1997.

## Principes pratiques de conception d'IHM



## Concevoir l'IHM : Principes pratiques

- **Les 5 règles d'or :**

- Une interface COHÉRENTE
- Une interface CONCISE
- Une interface RÉACTIVE
- Une interface FLEXIBLE
- Une interface STRUCTURÉE

## Une interface cohérente

- **Cohérence interne :**

- À utilisation identique, localisation identique
- À but identique, séquence de commandes identiques
- Si deux commandes ont des paramètres identiques, les placer dans le même ordre
- Terminologie (texte mais aussi iconographie)
- À sémantique identique, dénomination identique
- Respect du style d'interaction

## Une interface cohérente

- **Cohérence externe :**

- Choix des symboles (signifiant/signifié)
- Choix des termes
- Raccourci
  - ex: Ctrl-S ne doit pas vouloir dire parfois Sauver et parfois Supprimer
- La cohérence doit être globale au poste de travail (ex: respect du style Windows)
- Respect des standards de l'industrie

## Exemples d'incohérences

- Interface graphique

- Le "drag and drop" n'a pas la même sémantique si on fait glisser l'icône d'un fichier d'un dossier vers un autre du même volume (le fichier est déplacé) ou vers un autre volume (le fichier est dupliqué)
- Glisser l'icône d'une disquette dans la corbeille ne signifie pas que l'on veut effacer tout son contenu !

- Applications Microsoft

- Sous Excel, on peut sélectionner des zones non connexes et lancer une commande qui s'appliquera sur toutes les zones sélectionnées
- Sous Word, on ne peut pas sélectionner des zones de textes non connexes

- UNIX

- Incohérence lexicale pour l'opération "quitter" : une ligne commençant par un point pour terminer une édition, q pour quitter le mail, logout pour quitter la session
- Aucune logique dans les noms de commande : première lettre de chaque mot (cd pour Change Directory pwd pour Print Working Directory) ou première lettre des deux premiers mots et plusieurs lettres du troisième (grep pour Get Regular Expression, mkdir pour MaKe DIRectory)

## Une interface concise

- KISS : Keep It Simple and Stupid
- L'UTILISATEUR doit soutenir les tâches de l'utilisateur le plus simplement possible
- Eviter les fonctions et informations superflues



24 caractéristiques



7 caractéristiques

## Une interface concise

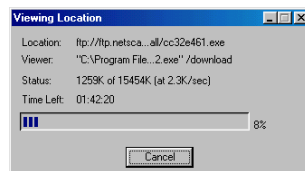
- Minimiser la charge mentale :
  - Mémoire à court terme très volatile et capacité réduite (7±2 éléments)
  - Il est plus facile de reconnaître que de se rappeler
  - Mémoire à long terme à très grande capacité mais l'effort de rappel est fatigant
  - Exemples :

## Une interface concise

- Combiner harmonieusement le bref et l'expressif. Mémoire à court terme limitée !
- Suivre des règles précises et naturelles pour les abréviations
- Donner la possibilité à l'utilisateur de créer simplement des macro-commandes
- Importation/ Exportation de données transparentes (par Copier/Coller ou autre)
- Utiliser des valeurs par défaut
- Masquer les opérations impossibles dans le contexte (griser les options d'un menu, etc)
- Pouvoir refaire et défaire
- Pouvoir appliquer une même opération sur différents objets

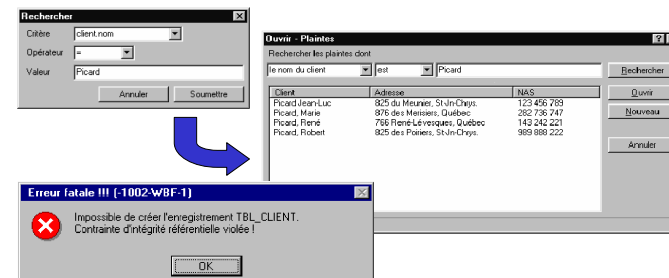
## Une interface réactive

- Le retour d'information (feedback) doit être immédiat et informatif : les variables psychologiques, centres d'intérêt de l'utilisateur, doivent trouver leur correspondant dans les variables physiques de l'image
- Informer pour rassurer, pour réduire la charge cognitive
- Efficacité, rapidité, fiabilité



## Une interface réactive

- Terminologie de l'utilisateur omniprésente



## Une interface réactive

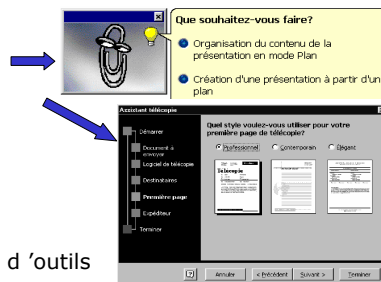
- Offrir des façons simples et évidentes pour terminer une opération ou sortir d'une fenêtre
  - Toute fenêtre d'application ou de document doit avoir un article de menu Fermer ou Quitter
  - Toute fenêtre de dialogue doit avoir un bouton « Annuler »

## Une interface flexible

- **Une interface adaptable : modifiable par intervention explicite de l'utilisateur**
  - Pouvoir modifier le lexique (adapter la terminologie, reformuler les messages, redéfinir les raccourcis) facilement, sans intervention d'un informaticien (édition de fichiers de ressources)
  - Pouvoir modifier les valeurs par défaut (fichiers de profil)
  - Représentations multiples des objets de l'interaction (ex : dans les logiciels de dessin, utilisation ou non d'une grille)
- **Une interface adaptative : qui se modifie automatiquement en fonction de l'activité de l'utilisateur**
  - Gestion d'un modèle de l'utilisateur (à ne pas confondre avec le modèle de l'utilisateur au sens de D. Norman) : utilisation de techniques d'IA

## Une interface flexible

- Pour les novices
  - Agents ou guide
  - Assistants (*wizard*)
  - Version *light*
- Pour les experts
  - Raccourcis clavier
  - Menus contextuels
  - Options avancées
  - Icônes dans la barre d'outils

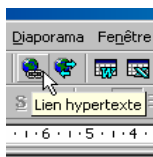
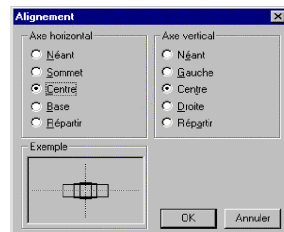


## Une interface structurée

- Réduire la complexité du fond (l'activité)
  - Organiser les fonctions en niveaux de complexité croissante (technique du scaffolding - étayage)
- Réduire la complexité de la forme (l'image)
  - Séparer
  - Hiérarchiser (navigation)
  - Reversibilité

## Une interface structurée

- Reconnaissance
- Regroupement selon le sens
- Listes et énumérations plutôt que codes
- Info-bulles pour les icônes et les abréviations



## Une interface structurée

### Prévenir les erreurs

- Mieux vaut prévenir que guérir !
- Inactiver les fonctionnalités que ne peuvent ou ne doivent être utilisées
- Indiquer le format de certains champs à l'aide d'info-bulles ou utiliser des filtres
- Utiliser des messages d'avertissement pour les opérations longues ou risquées
  - Attention aux estimations de temps folkloriques

## Une interface structurée

### Offrir un traitement simple des erreurs

- Etre prescriptif : indiquer ce qui ne va pas et comment y remédier
- Utiliser des exemples de solutions dans les messages
- Eviter les mots violents : illégal, fatal, sévère, avorté
- Eviter les points d'exclamation !!!
- Les messages doivent être auto-explicatifs (il ne faut pas avoir recours au manuel)



## Une interface structurée

### Ne rien cacher

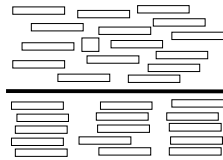
- Ne cacher aucune fonction derrière une séquence de touches magiques
- Tout résultat d'action doit être visible
- Tout état doit être visible



## Une interface structurée

### Représentation des données

- Objectifs :
  - Donner des vues synthétiques pour aider à la découverte et à la compréhension de collections de données
  - Présenter les résultats d'une requête
  - Présenter des relations entre différents éléments d'information
- Bon usage des dimensions graphiques : toute variable visuelle a ses particularités  
ex: utilisation de la grille



## Une interface structurée

### Techniques de visualisation

- **Organisation linéaire**
  - Liste traditionnelle (limitée par surface occupée et lenteur du parcours)
  - Défilement à grain variable
  - Mur fuyant
- **Organisation hiérarchique**
  - Arbre traditionnel limité en terme de nombre de nœud
  - Approche pavage (tree-maps)
  - Approche 3D



## Les stratégies médiatiques

- **Organisation des écrans :**
  - Travaux de Spérando : Fréquence oculaire

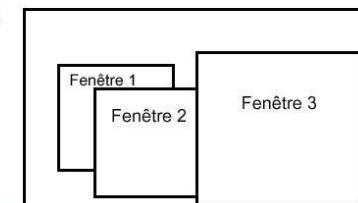
La plus consultée (alarmes ou conseils d'actions du SI)	?
?	La plus consultée (par Exemple, les commandes de l'interface)

[Spérando83]

## Multi-fenêtrage

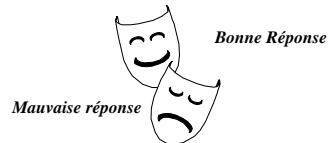
■ Multi-fenêtrage est une alternative à la structuration de l'information:

- ★ Utile en supervision de logiciels
- ★ Utile pour la mise en // d'Informations
- ★ Peu de fenêtres : 3 à 4
- ★ Gestion simple



## Les Icônes

- ICONES = Pictogrammes, objets-clés, symboles « parlants » dispensant de commentaire
- REPETITIFS = Repère pour identifier immédiatement le type d'info auquel il s'intègre
- Avoir cependant le sens de chaque bouton par simple survol (Texte)
- Attention à la disposition des boutons dans la barre de menu (e.g. ne pas mettre le bouton « écran suivant » à côté du bouton « Quitter »)

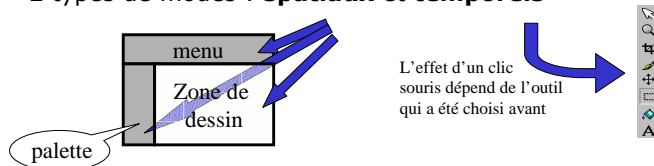


## Techniques et styles d'interaction

# Respect d'un style d'interaction indispensable

## Modes d'interaction

- Un mode est un état de l'interface dans lequel les actions de l'utilisateur sont interprétées par le système de manière homogène et différente des autres modes.
- 2 types de modes : **spatiaux** et **temporels**



## Modes d'interaction

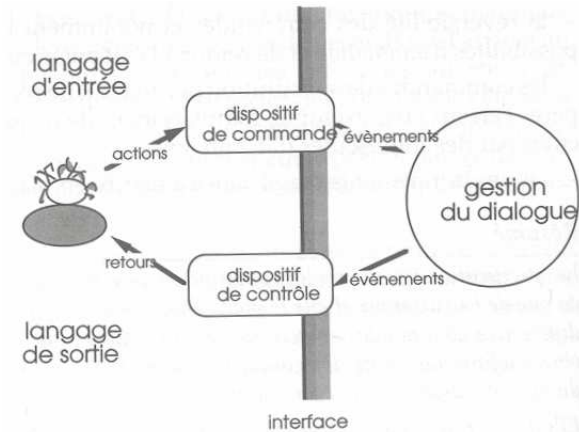
- **Micro-modes** :
  - mode temporel lié à une action physique continue de l'utilisateur, ex: *drag*
- Privilégier **mode spatial** et **micro-mode** car plus simples à mémoriser
- Si possible syntaxe "objet-action" plutôt que « action-objet »
  - Mais difficile pour création



## Dialogue Objet-Action

- **Donne le contrôle à l'utilisateur sans l'obliger à maîtriser un langage de commande**
- **Dialogue = Suite de transactions à 1 Transaction =**
  - Sélection d'un objet par pointage
  - Mise en œuvre d'une action sur l'objet
- **Avantages :**
  - Focalisation de l'utilisateur sur les objets visibles en permanence
  - Peu de mémorisation nécessaire, d'erreur de dénomination
  - Limitation du choix des actions en fonction du contexte (objet sélectionné)
- **Interfaces événementielle / Objets**

## Dialogue Objet-Action



## Styles d'interaction : Métaphores

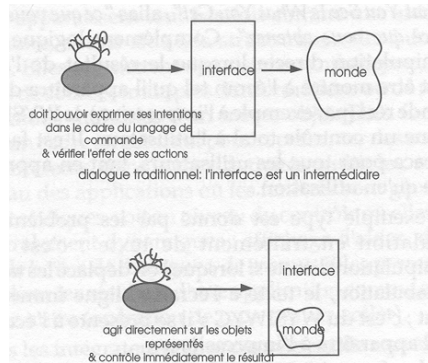
- Transfert d'une compréhension ou perception existante dans un nouveau contexte (objet, opérations, règles de manipulation)
- But : Accroître l'intuitivité et la convivialité
- Métaphore fonctionnelle : liée à la tâche à réaliser
  - Exemple : tableur à feuille de calcul
- Métaphore opérationnelle : lié au opérations disponibles
  - Exemple : Windows à bureau, Lecteur CD à Lecteur CD
- Métaphore organisationnelle : lié au emplacement/catégories des informations
  - Exemple : Annuaire à entreprise 3D, cartes

## Styles d'interaction

- **WYSIWYG : What You See Is What You Get**
  - Visualisation concrète des résultats des actions
  - Facile et efficace en apprentissage et utilisation
- **Navigation Hypermédia**
- **Manipulation directe**

## Manipulation directe (Schneiderman 1983)

- L'utilisateur a l'impression d'agir **directement** sur un monde qui s'apparente à un monde réel.



## Manipulation directe (Schneiderman 1983)

- **Principes :**
  - Représentation continue des objets
  - Utilisation d'actions physiques au lieu de syntaxe complexe
  - Opérations rapides, incrémentales et réversibles, dont les effets sur les objets sont visibles immédiatement: *feedback*
  - Apprentissage progressif
- **Nécessité d'un périphérique de localisation**

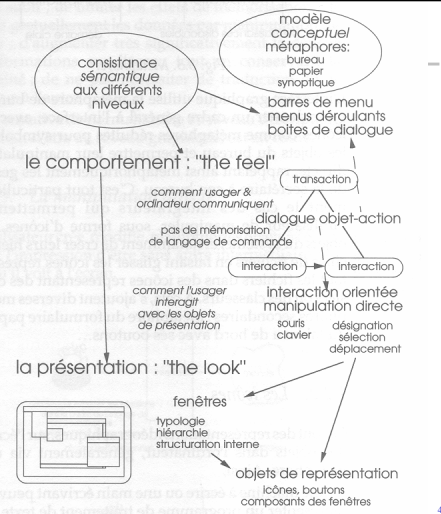
## Styles d'interaction : Look & Feel

- **Look = présentation = niveau lexical**
  - Objets de présentation : fenêtres
  - Objet de dialogue ou Interacteurs : menu, boîte de saisie, boutons, ...
- **Feel = comportement = dialogue = niveau syntaxique**
  - Interaction : souris, clavier, ...
  - Transaction : Suite d'interactions
- **Association Look & Feel = cohérence = niveau sémantique**
  - Signification des commandes

## Styles d'interaction : WIMP

- **WIMP : Windows, Icons, Menus, Pointer :**
  - Basée sur la métaphore du bureau
  - Icônes
  - Fenêtres 2D ou 2D 1/2
  - Menus déroulant ou popup
  - Boîtes de dialogue
  - WYSIWYG
  - Dialogue Objet-Action

## Interfaces WIMP



## Modèle d'interfaces en couches |Meinadier91]



## Choix de styles pour différentes tâches

- Saisie de valeurs
- Sélection d'un ou plusieurs objets parmi un ensemble
- Déclenchement de commandes
- Défilement de documents
- Spécifications de propriétés ou d'arguments
- Transformations graphiques

## Saisie de valeurs (par M.D.)

- Texte :
  - boîte de saisie (ligne unique, multiligne, filtrée)
- Quantité :
  - saisie clavier,
  - positionnement curseur,
  - incrémenter et décrémenter avec boutons,
  - liste...
- Position :
  - pointage et validation (clic).
  - Variations suivant opérations (rectangle, ligne brisée...)
- Tracé :
  - échantillonnage du pointeur

## Sélection d'un ou plusieurs objets

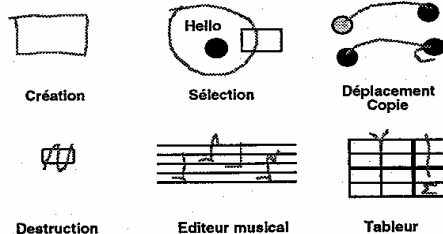
- **Dans un ensemble variable de choix :**
  - Directe : par pointage de la cible. Pour aider : feedback de la cible potentiel et gravité.  
Si sélection multiple : par groupe (rectangle, lasso) ou par ajout/retrait (ctrl-click).
  - Dans liste : par sélection de l'item.  
Si sélection multiple : par intervalle (premier, dernier) ou par ajout/retrait (ctrl-click).
  - Par nom : Si l'utilisateur connaît les identificateurs et qu'ils sont simples
  - **Parfois, on combine liste et nom : Au fur et à mesure que l'on saisie les premières lettres d'un nom, la liste se positionne.**

## Sélection d'un ou plusieurs objets

- **Dans un ensemble fixe de choix :**
  - Directe : par pointage de la cible.
  - Menu : Fixe ou surgissant. Hiérarchique ou circulaire  
Nombre de rubriques limité (7 +/- 2) sauf liste homogène (ex: polices) qui doit être ordonnée.  
Importance des *feed-back*, séparateurs et grisés.
  - Exclusif : Bouton radio
  - Binaire : Case à cocher, bouton 2 états

## Déclenchement de commandes

- Rubrique de menu
- Bouton de boîte de dialogue ou de palette
- Manipulation directe (ex: *Drag and Drop*)
- Double click
- Raccourci clavier (pour expert)
- Commande vocale
- Entrée gestuelle



## Défilement de documents

- Barres de défilement (ascenseur)
  - Attention inversion sens
  - Vitesse de défilement variable (ligne ou page)
- Défilement direct (roulette de souris type intellimouse, joystick)
- Défilement automatique (curseur en limite de zone)

## Spécifications d'arguments ou de propriétés

- Spécification d'arguments :
  - Directe (dans l'action)
  - Implicite (sélection en cours)
  - Explicite (boîte de dialogue)
- Boîte de dialogue (modales et non modales) :
  - Validation (OK), Annulation (Annuler)
  - Importance des valeurs par défaut
  - Parties optionnelles (à conserver dans l'état précédent)
  - Rubriques (onglets)
  - Boîtes de progression (dès que attente > 2 secondes...)
  - Boîtes de propriétés non modales

## Transformations graphiques

- Simplicité et confort
  - Poignées de manipulation directe
    - *déplacement*
    - *agrandissement*
    - *Rotation*
- Précision
  - Boîtes de dialogue
    - *ex : Angle d'une rotation, position X,Y*

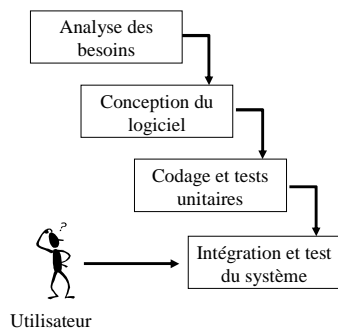
## Production de systèmes interactifs



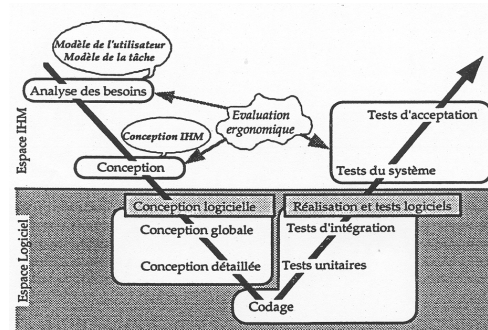
## Production de systèmes interactifs

- **Les enjeux**
  - 80 % du code des systèmes interactifs est consacré à l'interface utilisateur
  - Les **risques** d'une mauvaise interface :
    - Rejet pur et simple par les utilisateurs (technopathie)
    - Coût d'apprentissage (formation)
    - Perte de productivité
    - Utilisation incomplète (manque à gagner)
    - Coût de maintenance
    - Perte de crédibilité

## Méthode classique en cascade

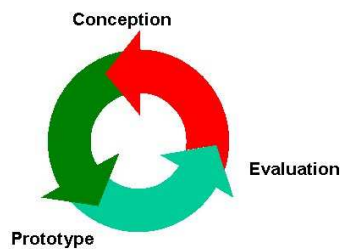


## Le modèle en V



## Un concept clé : l'Utilisabilité

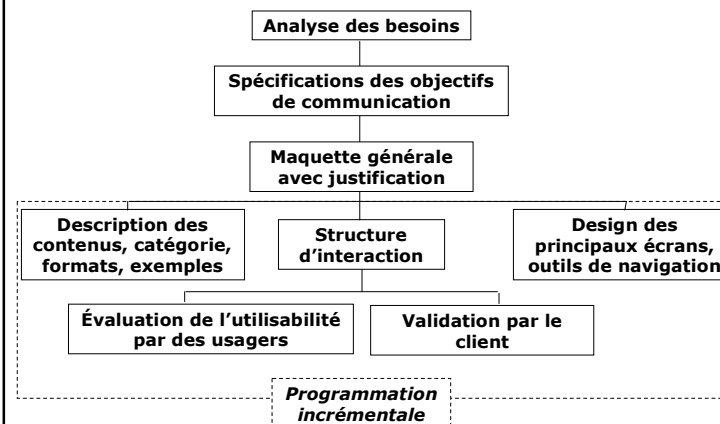
- Conception incrémentale
- Conception centré sur l'utilisateur



Il faut faire participer l'utilisateur dans toutes les phases amont :

**Participative design**

## Méthodologie de conception



## Analyse des besoins

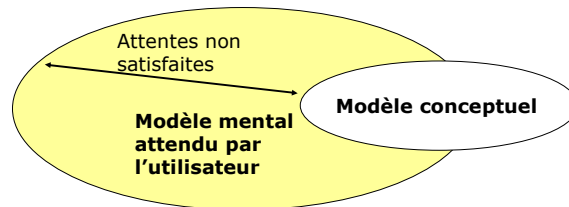
- **Comporte 2 facettes par souci de prise en compte des éléments spécifiques de l'interaction homme-machine** [Buisine1999] :
  - · le modèle fonctionnel permettant l'identification des fonctionnalités du système et des flux d'informations entre celles-ci,
  - · le modèle de l'activité prescrite qui complète le modèle fonctionnel par une description de l'activité telle qu'elle est prévue avec le futur système
- **En phase d'analyse et de spécification du système, ces modèles deviennent :**
  - · le modèle fonctionnel dont la finalité est identique au modèle fonctionnel de la phase d'analyse de besoin, mais qui est orienté par les fonctionnalités de la solution technique,
  - · le modèle d'interaction qui complète le modèle fonctionnel système par les éléments propres à l'interaction homme-machine (type de dialogue, type de modalité, ...)

## Modèles conceptuels

- Représentation mentale qui dépend de la connaissance acquise (évolue avec expérience)
- 3 modèles conceptuels :
  - de l'utilisateur à propos du système
  - du concepteur à propos du système et de l'utilisateur
  - du système à propos de l'utilisateur
- Correspondance entre :
  - les variables psychologiques de l'utilisateur et
  - les variables physiques du système

## Modèles conceptuels

Dualité modèle du concepteur – modèle mental de l'utilisateur



- **Objectif** : réduire les distances entre les modèles (Théorie de l'action de Norman)

## Méthodologie de conception

1. Définir le modèle de l'utilisateur
2. Définir le modèle de la tâche
3. Définir des concepts informatiques
4. Définir le modèle de l'interaction

- Au moment de la **spécification détaillée**, on doit disposer :
  - Des 3 modèles
  - De scénarios d'utilisation

## 1. Définir le modèle de l'utilisateur

---

- **Objectifs :**
  - Identifier les caractéristiques pertinentes de(s) l'utilisateur(s)
  - Réduire les distances d'exécution et d'interprétation
- **Pour cela :**
  - Parler aux utilisateurs n'est pas un luxe, c'est une nécessité
  - Il faut se concentrer sur l'utilisateur le plus tôt possible et de façon continue
  - Il faut faire participer l'utilisateur à la conception

## 1. Définir le modèle de l'utilisateur

---

- **Problèmes :**
  - Choix des utilisateurs représentatifs
  - Accès aux utilisateurs
  - Omission du contexte d'utilisation
- **Représentation :**
  - Données générales biométriques et sociales
  - Connaissance dans le domaine de l'application et en informatique

## 1. Définir le modèle de l'utilisateur

---

- **Techniques :**
  - Classification générale si utilisateur non disponible
  - Interview
  - Questionnaire (6 à 12 utilisateurs représentatifs)
    - ~ Degré de confiance, "je ne sais pas"
    - ~ Poser plusieurs fois la même question de façon différente pour comparer
    - ~ Eviter les réponses prévisibles

## 2. Définir le modèle de la tâche

---

- **Tâche :** But + Procédure pour l'atteindre
- **Procédure :** ensemble de sous-tâches liées par des relations de composition et des relations temporelles
- **Tâche élémentaire :** tâche décomposable en actions physiques (opération d'E/S)



## 2. Définir le modèle de la tâche

### • Méthodes :

- Observation directe (enregistrement avec 2 observateurs) en laboratoire ou sur le terrain
- Entretien (ouvert, dirigé, incident critique)
- Groupe de travail : rassembler utilisateurs et concepteurs (brainstorming, scénario)

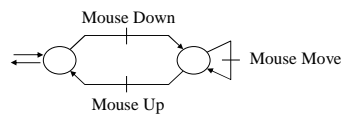
## 2. Définir le modèle de la tâche

### • Résultats attendus :

- Identifier les variables psychologiques
- Construire la hiérarchie de tâches jusqu'aux tâches élémentaires
- Décorer chaque tâche : préconditions, postconditions, fréquence, complexité, criticité, acteur responsable (utilisateur ou système) ...
- Penser aux exceptions
- Evaluation avec utilisateur

## Modélisation des tâches

- Les tâches de l'utilisateur peuvent être décomposées en sous-tâches élémentaires.
- On pourra ensuite modéliser une tâche complexe sous forme d'une machine à états où chaque transition est une action élémentaire.
  - Utilisation de formalismes comme réseau de Pétri



## 3. Définir des concepts informatiques

### • Objectif :

- Etablir correspondance entre les concepts du domaine et les concepts informatiques
- Distinguer parmi les objets informatiques ceux qui dépendent du domaine et ceux qui sont de l'IHM

### • Sélectionner les services généraux :

- Aide
- Défaire-refaire
- Couper-copier-coller
- Valeurs par défaut

## 4. Définir le modèle d'interaction

Voir le cours sur les styles et modes d'interaction

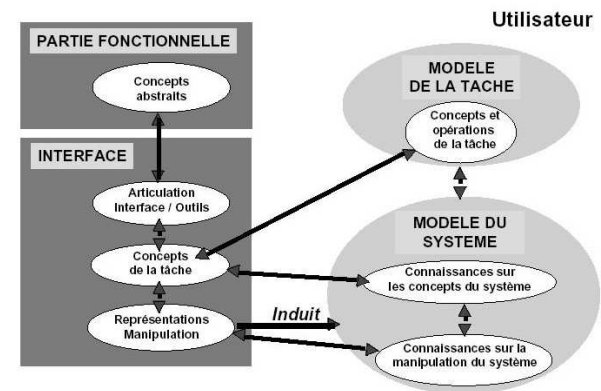
## Modèles d'Architectures



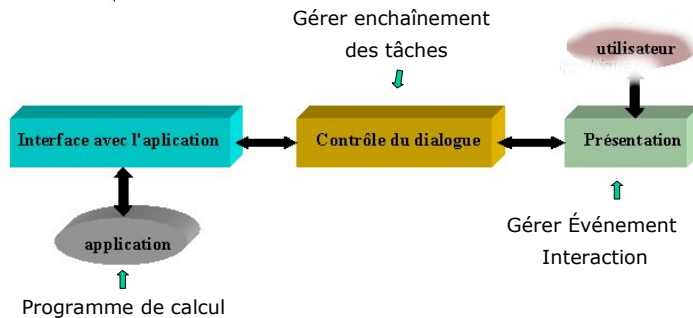
## Pourquoi des modèles d'architectures?

- 1 Conception d'IHM
  - o difficile à itératif
  - o Itérativité à modifiabilité du logiciel
- 1 Savoir faire artisanal insuffisant
- 1 Complexité croissante des IHM
- 1 Outils de développements imparfaits
  - o boîtes à outils : niveau d'abstraction trop bas, absence structuration
  - o squelettes d'appli : reverse engineering nécessaire pour réutilisation
  - o générateurs d'IHM : syndrome de l'ABS (faux sentiments de sécurité)
- 1 Séparation noyau fonctionnel / interface :
  - o Noyau fonctionnel -> code du domaine applicatif
  - o Interface -> filtre entre utilisateur et noyau

## Modèles d'architecture



## Modèle de type langage : SEEHEIM [1985]



## SEEHEIM : un modèle de type langage

- La machine et l'utilisateur sont deux entités qui parlent entre elles
  - Niveau sémantique : concepts et savoirs-faire du domaine  
Ex : la température, augmenter la puissance du radiateur...
  - Niveau syntaxique : phrases(ou images) correctes représentant les objets et les actions que l'on peut faire sur eux  
Ex : image d'un thermomètre, agir sur un bouton pour augmenter la puissance du radiateur
  - Niveau lexical : vocabulaire de base qui permet de construire les phrases  
Ex : :caractères du clavier, sons élémentaires, primitives graphiques

## SEEHEIM : La présentation

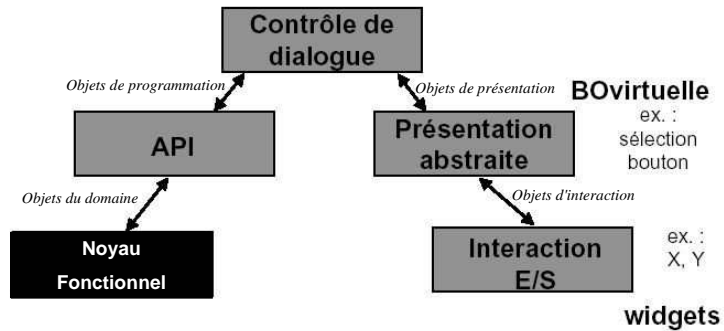
- Elle est responsable de la représentation physique du système, perceptible par l'utilisateur. Elle assure l'analyse lexicale du langage d'interaction
- Elle lit les données en provenance des dispositifs d'entrée.
- Elle traduit les unités lexicales d'entrée en expressions symboliques compatibles avec la représentation informatique interne et qu'elle transmettra au contrôleur de dialogue
- Elle reçoit les expressions abstraites en provenance du contrôleur de dialogue et les traduit en unités lexicales de sortie (textes, sons, etc)

## SEEHEIM : Le contrôleur de dialogue

- Il est responsable de l'analyse syntaxique du langage d'interaction
- Il assemble les unités reçues de la présentation en phrases qui correspondent à des requêtes ou des données que l'utilisateur veut envoyer à l'application
- Inversement, il ventile les phrases reçues de l'interface avec l'application vers les éléments spécialisés de la présentation
- Il organise aussi la structure du dialogue (création des contextes d'interaction)
- **L'interface avec l'application :**
  - Elle définit la vue que le contrôleur a de l'application. Elle traduit les phrases émises par le contrôleur dans le formalisme interne de l'application

## Modèle en couches

- **Modèle ARCH [1990]**



## Modèle ARCH [1990]

- ARCH : un métamodèle
  - générer des modèles d'architecture
  - ex. principe Slinky : module les 2 adaptateurs
- Modifiabilité
  - réparation sémantique
  - protocole de haut niveau
- Portabilité IHM
  - boîte à outils virtuelles
- Extensibilité
  - Objets d'interaction dédiés

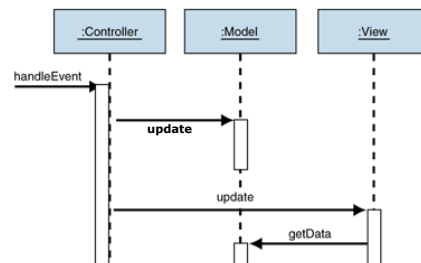
## Le modèle MVC (Model-View-Controller)

Le *Model-View-Controller (MVC)* sépare les données de la présentation grâce à 3 classes [Burbeck92]:

**Model** gère les données, répond aux requêtes de demande d'état sur les données (venant de view), et répond aux requêtes de mises à jour (venant de controller).

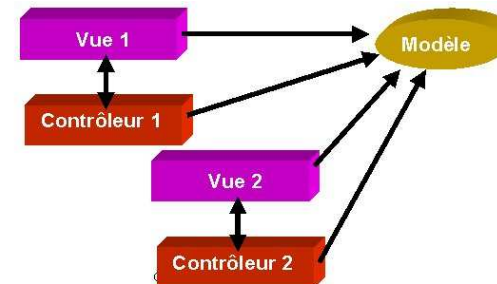
**Controller** interprète les entrées souris et clavier, informe le model et/ou view des changements à faire.

**View** affiche les informations

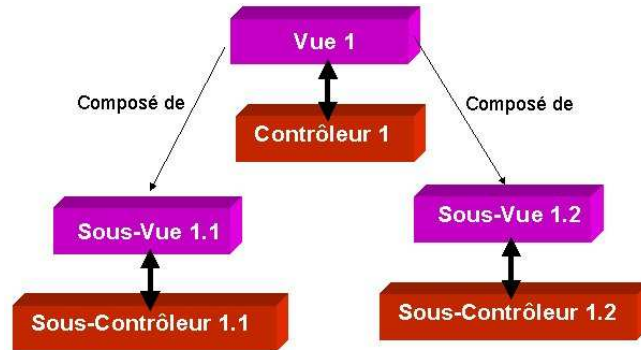


## Le modèle MVC (Model-View-Controller)

- Chaque entité interactive (widget) est une "triade" d'objets **SmallTalk** : un modèle, une vue, un contrôleur
- Connexions explicites entre les membres d'une triade :



## Hierarchie visuelle et hiérarchie de contrôle en SmallTalk

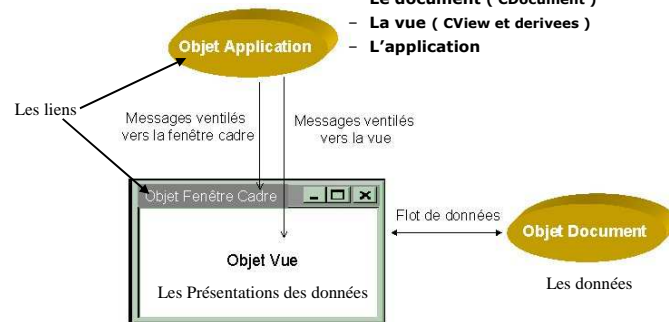


## Contrôle général de l'interaction en SmallTalk

- Un agent : 3 objets smalltalk
- Hiérarchie de models, de views, de controllers
  - Model : la compétence abstraite de l'agent (NF)
  - View : rendu en sortie
  - Controller : comportement en entrée
- Le gestionnaire de processus ordonnance tous les processus du système : gestion dynamique de la mémoire, gestion des fenêtres, etc.
- Le gestionnaire de fenêtres ordonnance les processus liés aux contrôleurs. C'est lui qui décide quel est l'objet interactif courant
- A chaque fois qu'un Modèle est modifié, il le notifie à toutes les vues qui le concernent grâce à un dictionnaire

## L'architecture Document-View de MFC

- 5 objets essentiels:
  - La fenêtre cadre ( CFrameWindow et variantes )
  - Le modèle de document ( Cdoc-template )
  - Le document ( CDocument )
  - La vue ( CView et derivees )
  - L'application



## L'architecture Document-View de MFC

- Quel est le rôle du document ?
  - Contenir l'information : ou bien parfois seulement contenir un pointeur vers l'information, comme par exemple l'URL d'un site Internet, ou un DSN pour une base de données.
  - Transformer les données
  - Sérialiser les données : Sauvegarder et recharger les données.
  - Synchroniser : Le document connaît **ses vues**. Il les synchronise, envoie des commandes de mise à jour, il centralise les modifications et redistribue les ordres de mise à jour.
  - **Traiter** les commandes : Les commandes qui modifient directement les données doivent être traitées par le document

## L'architecture Document-Vue de MFC

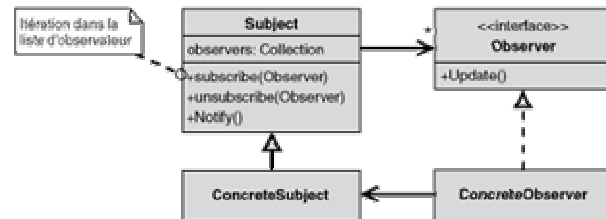
- **Quel est le rôle de la Vue ?**
  - Gère IHM
  - peut être considérée comme un périphérique d'entrée (clavier, souris)
  - peut être considérée comme un périphérique de sortie. La vue est alors chargée d'afficher les données modifiées
- **A quoi sert la fenêtre cadre ?**
  - Simplement à aider l'utilisateur à identifier facilement les vues.
    - les fenêtres cadres ne sont pas toujours nécessaires.
    - une seule fenêtre cadre en SDI, celle de l'application. Mais noter que l'on peut disposer de multiples vues en SDI, et en changer facilement en utilisant les splitters ou encore les onglets.
  - fournir un point d'ancrage pour les outils tels que barre de statut et autres "toolbars"

## L'architecture Document-Vue de MFC

- **Et le rôle du "document template" ?**
  - C'est la colle, le liant.
  - Le Doc-template fait le lien entre type de document, type de vue, type de fenêtre cadre et les ressources qui y sont associées.
  - il est nécessaire de choisir entre CSingleDocTemplate (SDI) et CMultiDocTemplate. (pour MDI)
- **Permet aussi de définir simplement :**
  - Une ressource pour une icône.
  - Une ressource pour un menu. En mode MDI, MFC changera automatiquement de menu lors de l'activation de telle ou telle fenêtre cadre.
  - Une ressource "string". C'est la concaténation de 7 sous-chaines séparées par "\n". (le titre de la fenêtre cadre. le nom du document par défaut. ...)

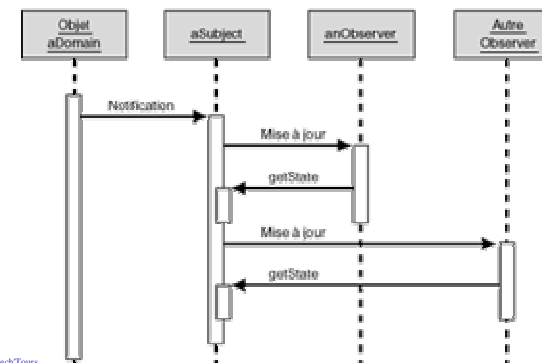
## Modèle Observer/ Observable

- **Subject = Données / Abstraction = Classe Observable**
- **Classe Observer = Observateur qui veut être informé des modifications des données observables**
- **Interfaces Java**



## Modèle Observer/ Observable

- Notification par méthode Push ou Pull



## Systemes Multi-Agents

- La Programmation objet s'appuie sur la distribution des connaissances (attributs) et des compétences (méthodes) au sein d'objets.
- Un programme n'est plus une entité particulière - c'est une organisation d'entités de plus en plus faiblement couplées.
- De nouvelles méthodes de conception en génie logiciel
- Un changement de perspectives de la notion de programmes à celle d'organisation.

## Agent

- Un agent possède trois types de connaissance :
  - connaissances du domaine (ou expertise),
  - connaissances de contrôle (ou comportement),
  - connaissances de communication et interactions
- C'est sur les formalismes, les méthodes de conception et de réalisation de ces trois types de connaissances que se concentre la recherche actuelle en IAD.

## Un système multi-agents

- Un environnement E,
- Un ensemble O d'objets situés dans E :
  - passifs,
  - perçus, créés, détruits et modifiés par les agents.
- Un ensemble A d'agents (A inclus dans O), entités actives du système,
- Un ensemble R de relations unissant des objets entre eux (et donc aussi des agents ...),
- Un ensemble Op d'opérations permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O.
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification (=> ce sont les lois de l'Univers E).

## Le but d'un SMA

- Faire fonctionner ensemble un ensemble d'agents pour :
  - résoudre des problèmes,
  - accomplir des tâches fonctionnelles,
  - simuler des systèmes existants.
- Ce n'est pas forcément la décomposition d'un problème en Agents qui est difficile, mais plutôt la composition de leurs actions individuelles :
- Les méthodes de décomposition (Conception OO) existent, même s'il faut parfois les particulariser.

## Composition d'Agents

- Les méthodes de composition d'agents passent par l'analyse et l'implémentation :
  - des mécanismes de coordination,
  - des mécanismes de collaboration,
  - des mécanismes de coopération.
- Ces mécanismes sont mis en oeuvre par :
  - des techniques de communication,
  - des techniques de partage de tâches,
  - des techniques de partage de connaissances,
  - des techniques de partage de plans (et objectifs).

## Deux Pôles Extrêmes

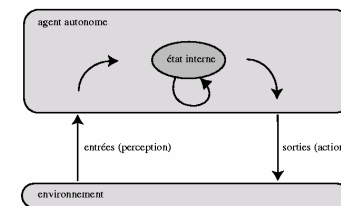
- Deux grandes classes de mécanismes existent :
  - **systèmes cognitifs** (capacité de raisonnement et de représentation des agents) - métaphore sociologique,
  - **systèmes réactifs** (minimise l'apport individuel de chaque agent) - métaphore biologique et éthologique.  
=> **Vie Artificielle**

## SMA Cognitifs

- Les SMA Cognitifs ont pour origine le désir de faire coopérer des Systèmes Experts :
  - représentation explicite, mémoire du passé,
  - planification, engagement,
  - mode d'organisation social,
  - peu d'agents (2 à 20).
- Chaque Agent possède (en général) :
  - une base de connaissance et une expertise du domaine,
  - une base de connaissance et une expertise du contrôle de son comportement,
  - une base de connaissance et une expertise de communication,
  - une base de connaissance et une expertise de représentation du monde et des autres agents,
  - une base de connaissance et une expertise de coopération entre agents.

## SMA Réactifs

- Pas de représentation ni de mémoire du passé (seulement un état courant),
- Stimulus-Réponse,
- Mode d'organisation biologique,
- Beaucoup d'agents (>100)



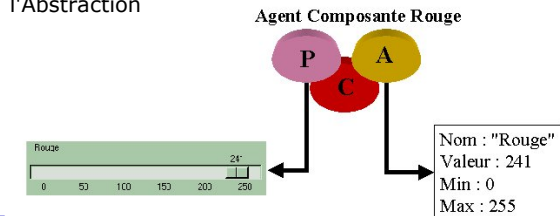


## Multi-Agents & IHM

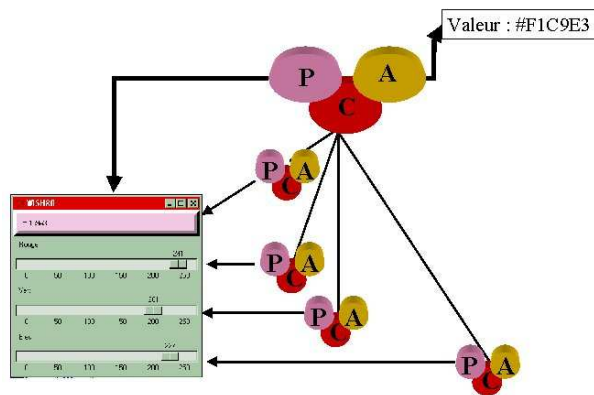
- Un système interactif : une collection d'unités de calculs spécialisées (agents..)
- Un agent IHM (Interacteur) :
  - a un état
  - a une expertise
  - est capable d'émettre et de réagir à des événements
- Intérêt :
  - Modularité : conception itérative
  - Distribution : mise en œuvre de collecticiels
  - Parallélisme : dialogue à plusieurs fils
  - Correspondance avec l'approche à objets
  - Encapsulation : l'agent seul modifie son état
  - Réutilisation : héritage, composition...

## Un modèle multiagent : PAC [Coutaz 87]

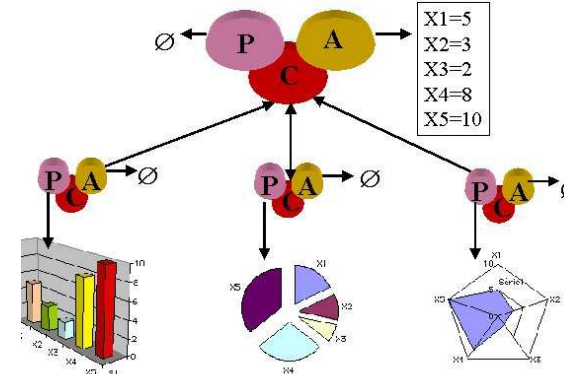
- Un agent PAC se caractérise par :
  - Une Présentation qui définit le comportement de l'agent perceptible à l'utilisateur
  - Une Abstraction qui contient la représentation interne de l'agent (attributs et méthodes)
  - Un Contrôleur qui assure le lien entre la Présentation et l'Abstraction



## Agent composé: Réglage de couleur



## Agent composé virtuel : plusieurs vues d'une même abstraction



### Problème de mise en œuvre

- Adéquation parfois difficile entre un modèle théorique et les outils de développement
- Ex 1 : les outils de présentation type CD-ROM interactif sont très performants sur le plan de la présentation et du contrôle mais sont souvent assez pauvres sur le plan de l'abstraction (Director et Toolbook)
  - Dissémination de l'abstraction dans la présentation

### Problème de mise en œuvre

- Ex 2 : les outils de programmation d'applications interactives complexes sont très performants sur le plan de l'abstraction, mais le contrôle est laissé à la discrétion du programmeur et il faut au moins 6 mois d'expérience pour maîtriser les boîtes à outils de présentation (type Visual C++, Borland C++, Java)
  - Apprentissage difficile
  - Vitesse de développement limitée

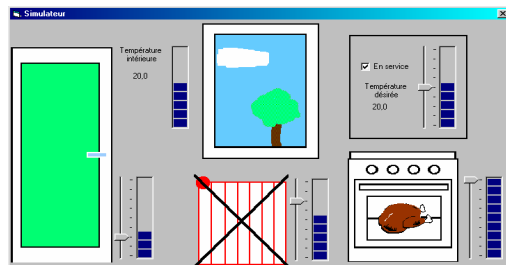
### Problème de mise en œuvre

- Ex 3 : les outils de programmation d'applications interactives simples constituent généralement un bon compromis (Visual Basic, Delphi)
  - Il faut s'imposer des règles de programmation pour ne pas tomber dans la facilité qui nuit à la maintenance et à l'efficacité
  - Utiliser programmation orientée objet chaque fois que c'est possible
  - Découpage en modules clairement identifiés

### Construction d'un simulateur de domotique

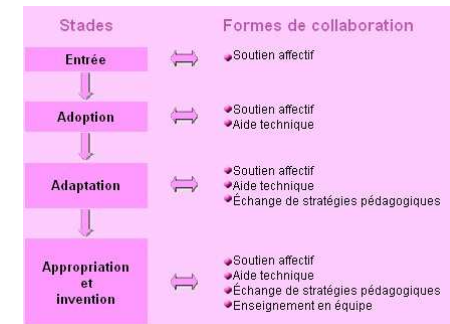
- Cahier des charges :
  - Contrôle - commande des appareils électroménagers d'une cuisine :
    - Four + radiateur + porte (sur extérieur) + ...
  - Mode de fonctionnement Manuel ou automatique (radiateur sous contrôle)
  - Simulateur pour validation de l'interface et des algorithmes de contrôle
    - Lois thermodynamiques simplifiées
    - Contributions calorifiques + pertes ...

## Une proposition



## Stratégie d'implantation

- Pour un succès : Tiré de [Dusfrene2000]
  - Laisser du temps pour s'adapter
  - Créer un climat propice à l'apprentissage



## Plasticité des Interfaces, Mutlimédia, Hypermédia



## Systèmes adaptatifs

- Types d'adaptation des interfaces (Edmonds, 1981)
  - ∅ Interface adaptée
  - ∅ Interface adaptable
  - ∅ Interface adaptative

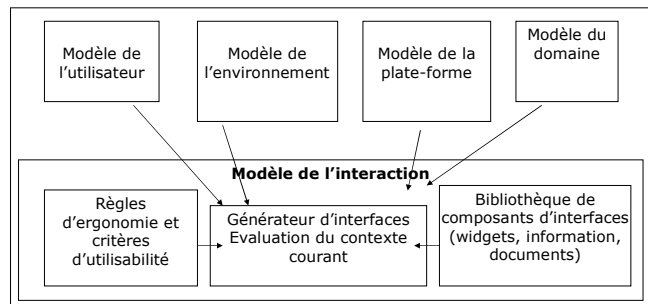
## Interfaces adaptatives

- Mise en œuvre :
  - Technologie permettant l'adaptation :
    - Hypermédia = XML + XSL + XSLT
    - Autre = Langages évolués (java, C, ...)
  - Modèle de l'utilisateur dynamique et précis !
- Notion de plasticité des IHM :
  - Adaptation à l'utilisateur mais aussi
  - au matériel disponible : périphérique d'E/S
  - au Système d'Exploitation, ...

## Interfaces adaptatives

- Modèle de l'utilisateur :
  - Ce qu'il sait : connaissances acquises
  - Ce qu'il est : profil cognitif (modèle de Hermann, ...)
  - Ce qu'il aime : ces préférences, ces goûts, ...

## Architecture des systèmes adaptatifs



## Contrôle de l'adaptation

§ Le Qui : initiateur de l'adaptation

	Système	Utilisateur
Initiative	X	
Proposition	X	
Décision		X
Exécution	X	

Auto-adaptation contrôlée par l'utilisateur

	Système	Utilisateur
Initiative	X	X
Proposition	X	
Décision	X	
Exécution	X	

Auto-adaptation

	Système	Utilisateur
Initiative		X
Proposition		X
Décision		X
Exécution	X	X

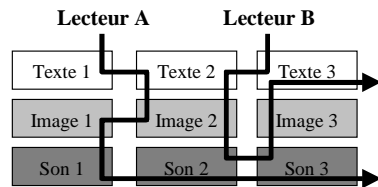
Adaptation

§ Le Quand : temporalité de l'adaptation

§ Le Comment : réalisation de l'adaptation

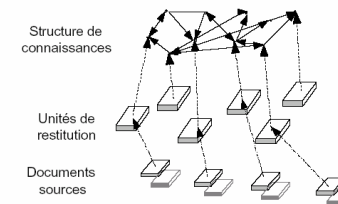
## IHM & Hypermédia

- Un **hypertexte** est un **hypertexte** dans lequel les informations enregistrées ne sont pas uniquement textuelles.
- **Intérêt d'un hypertexte** :
  - Chaque lecteur a son propre parcours.



## IHM & Hypermédiás

- Page Web = Document
- Reconstruire le sens [Nanard99] :
  - Distance opératoire
  - Distance articulatoire
  - Distance sémantique = évaluation, comprendre, rhétorique, chemin de lecture, risque d'incohérence



## IHM & Hypermédiás

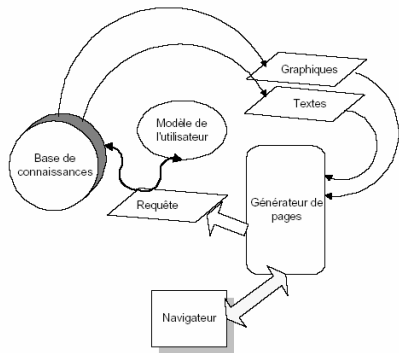
- Distance sémantique
  - évaluation, comprendre : comment énoncer les informations ?
  - rhétorique : qui lit ? pourquoi ?
  - L'art d'intéresser : page d'appât, ...
  - chemin de lecture, risque d'incohérence

## Comment lit-on une page Web ?

- **On ne lit pas une page web, on l'a parcourt (scan)**
- Utiliser :
  - Des mots surlignés (liens, couleurs, taille, police)
  - Des sous-titres significatifs
  - Des listes pointées
  - UNE SEULE IDÉE PAR PARAGRAPHE
  - Une style "pyramide inversée" : on commence par la conclusion
  - Supporter la motivation : pour éviter le survol/abandon
- Augmenter la crédibilité d'une page :
  - Graphiques de bonne qualité
  - Écriture (grammaire, orthographe) de qualité
  - Liens vers d'autres sites
  - Éviter à tout prix le style racoleur et les professions de foi

## Hypermédias adaptatifs

- **Définition :**  
Hypermédia capable d'adapter dynamiquement le modèle de navigation et le contenu des pages présentées en fonction du profil de l'utilisateur

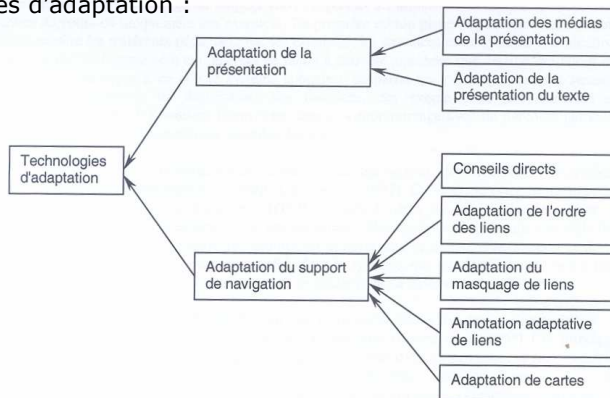


## Hypermédias adaptatifs

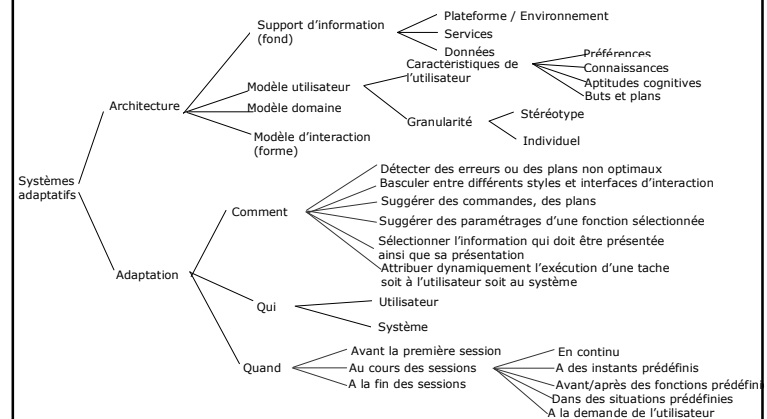
- **L'adaptation**
  - Du contenu :
    - forme textuelle, image, son, ... (ou suivant périphérique utilisé)
  - De la navigation :
    - Guidage direct (sequentialité, disparition de la navigation)
    - Ordonnancement de liens (suivant intérêt - cf moteur de recherche)
    - Masquage de liens (vers des pages trop simple ou trop complexes par exemple)
    - Annotation des liens : bulle comme pour icône mis par le concepteur ou par l'utilisateur lui-même à système Nestor)
- **Problème :** désorientation car non stabilité du contenu des pages

## Hypermédias adaptatifs

Types d'adaptation :



## Résumé



## Le multimédia

### • Définition

- Un **média** est un mode de représentation de l'information clairement caractérisable.
- Il peut-être :
  - **discret** (indépendant du temps) :
    - graphique,
    - image fixe,
    - texte (+/-),
  - **continu** (dépendant du temps) :
    - son,
    - images animées.
- Le **multimédia** (en informatique) : réunion sur un même support de plusieurs média.

## Intérêts du multimédia

### • Pourquoi le multimédia ?

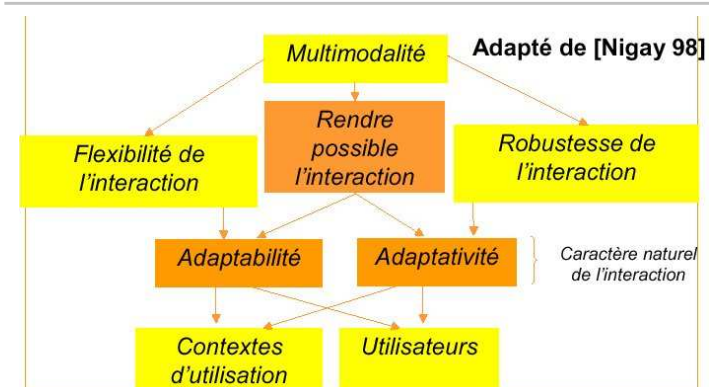
- L'homme par essence et par ses sens est multimédia.
- Des études portant sur l'**interactivité** montrent que l'être humain ne retient que :
  - 10 % de ce qu'il LIT
  - 50 % de ce qu'il VOIT et ENTEND
  - 70 % de ce qu'il VOIT et ENTEND et FAIT



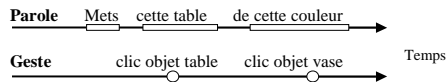
## Le multimédia

- Posent des problèmes de :
  - dépendance entre média,
  - synchronisation (spatiale, temporelle et sémantique).
- La prise en compte de l'utilisateur :
  - **Adaptabilité** : adaptation statique des préférences d'usages, capacités perceptives, ...
    - Pb = délais nécessaire pour l'adaptation, nouveaux périphériques, ...
  - **Adaptativité** : adaptation dynamique au cours de l'interaction
    - PB = Une seule interaction flexible qui s'adapte ?

## La multimodalité



## Difficulté majeure : la fusion



### • Critères d'intégration :

- Proximité temporelle
- Complémentarité sémantique
- Complémentarité par attribut
- Incompatibilité des modalités
- Ordonnement des références
- Complétude des événements utilisateur

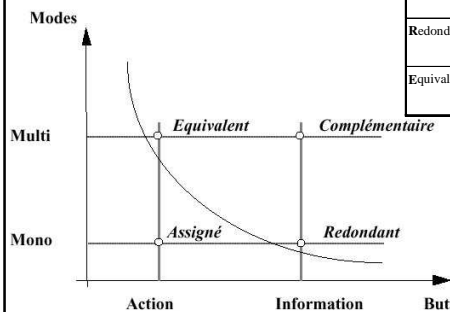
## Multimodalité en sortie

### • Problème de la multimodalité en sortie

- Génération + Présentation
- Quelles modalités choisir ? Critères :
  - Conditions d'utilisation (poste de travail et environnement)
  - Caractéristiques de l'utilisateur
  - Nature de la tâche

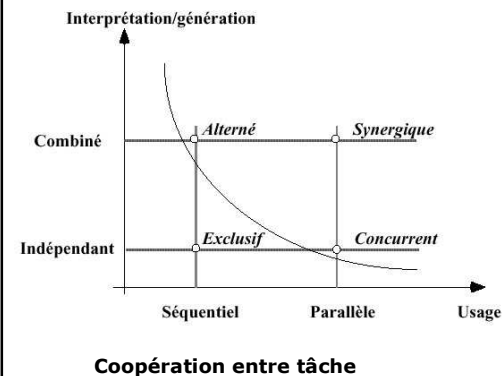
## Mode de coopération : Propriété CARE

### Coopération entre modalité



Coopération	Condition	Propriété
Complémentarité	Usage conjoint pour faire sens	Usage déictiques "mets ça ici"
Assignment	Absence de choix	Rapidité, efficacité
Redondance	Usage simultané pour action	Accroissement de la robustesse
Equivalence	Usage d'une seule modalité pour action	Souplesse dans le choix d'expression

## Modes de coopération : Propriété CASE



- Concurrent : 2 tâches distinctes en parallèle, pas de coréférence
- Alterné : entrelacement temporel
- Synergie : tâches en parallèle avec coréférence
- Exclusif : une seule tâche



## Multimodalité : Qq mythe (S. Oviatt)

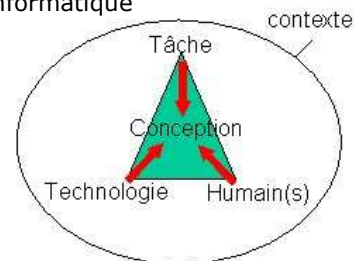
- Ce n'est pas parce qu'une application est multimodale que les utilisateurs vont utiliser la multimodalité
- Multimodalité <> parallélisme
- Les erreurs sur un mode ne sont pas compensées sur un autre
- Les utilisateurs n'organisent pas leur modalité de la même manière
- Les modes ne sont pas équivalents
- Un système multimodal n'est pas plus efficace qu'un autre

## Facteurs Humains en IHM



## Composants de l'interaction Homme-Machine

- l'utilisateur
- accomplit une tâche
- dans un contexte particulier : organisation
- en utilisant un système informatique



## Utilisateur : facteurs à prendre en compte

- Physiques et physiologiques : taille, poids, droitier/gaucher, acuité visuelle, dextérité, fatigabilité,...
- Psychologiques : vitesse d'apprentissage, capacité de mémorisation, curiosité,...
- Socio-culturels : niveau d'étude, sexe, origine sociale, pays,...
- Expérience professionnelle : niveau de compétence dans la tâche

## La tâche : facteurs à prendre en compte

- La tâche est-elle répétitive , peu sensible aux changements de l 'environnement ?
- Tâche régulière, occasionnelle, unique ?
- Le temps est-il un facteur critique ?
- Quels types de savoirs et de savoir-faire sont-ils requis ?
- Des situations critiques peuvent-elles survenir (sécurité) ?
- L 'utilisateur doit-il porter des vêtements de protection ou utiliser des périphériques d 'E/S inhabituels (environnement dangereux, handicap) ?
- L 'utilisateur accomplit la tâche seul ou avec d 'autres ?
- L 'utilisateur accomplit plusieurs tâches en parallèle ?
- Le travail effectué par l 'ordinateur est-il fiable ? Quel est le partage des décisions entre l 'homme et la machine ?

## L'organisation : dimension sociale

- En amont :
  - L'analyse des besoins dans contexte organisationnel, social, culturel, affectif...
  - Les usagers sont rarement dans l'isolement, ils communiquent à propos de leurs usages et inventent de nouveaux usages.
  - Constitution de communautés de pratique
- En aval :
  - Evaluation : de l'utilisable, de l'utile, de l'acceptable
  - Etude de la vie du système et des stratégies de réussite : rôle des autres acteurs, poids de la hiérarchie, notion de bénéfice partagé par tous les acteurs.

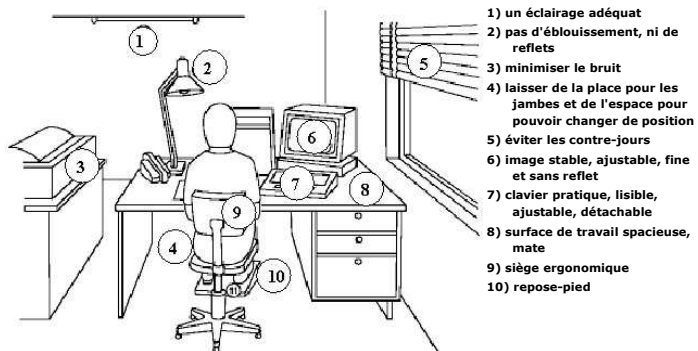
## L'organisation : facteurs à prendre en compte

- Attribution des tâches : qui fait quoi ? Comment les résultats sont-ils contrôlés ?
- Dépendances entre tâches (workflow)
- Partage du pouvoir et de l'influence
- Existence d'un support technique (service informatique interne, collègues...)
- Politique de formation
- Gestion du personnel
- Contraintes économiques

## Systeme informatique : facteurs à prendre en compte

- **Contraintes matérielles**
  - capacité mémoire et puissance de calcul, qualité des périphériques d'E/S
- **Contraintes logicielles**
  - Système d'exploitation, applications disponibles
- **Organisation des machines**
  - Machines isolées, en réseau (intranet, extranet,...)

## L'environnement de travail



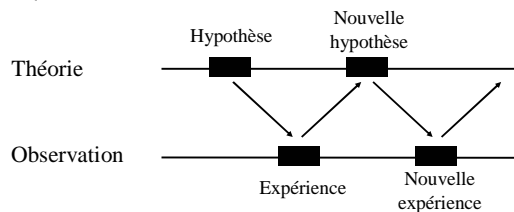
## Un concept clé : l'Utilisabilité

- Réaliser des systèmes (informatiques ou non) permettant à l'utilisateur d'accomplir ses tâches :
  - De façon efficace (« effectively »)
  - Avec une bonne productivité (« efficiently »)
  - En toute sécurité (« safely »)
  - En prenant plaisir à le faire (« enjoyably »)
  - En apprenant rapidement à utiliser le système

## Psychologie cognitive et IHM

- La psychologie cognitive offre des modèles théoriques du fonctionnement de l'esprit humain pour tout comportement mettant en œuvre des connaissances.

- " Plans " et analyse de tâches
- Très quantitatif



## Psychologie cognitive et IHM

- Modéliser
  - Modèles expérimentaux / théoriques
  - Modèles prédictifs / explicatifs
  - Modèles de performances / comportements
    - activités sensori-motrices (perception...)
    - activités mentales (acquisition de connaissances, raisonnement...)
- Comprendre et détecter
  - Ce qu'il faut faire / ne pas faire
- Expliquer / Construire
- Evaluer

## Psychologie cognitive et IHM

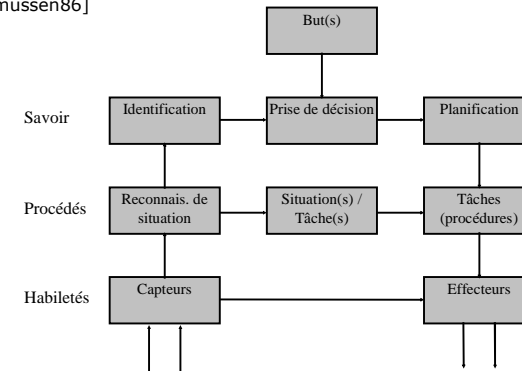
- Qu'est ce que la connaissance ?
  - Elle ne se transmet pas.
  - Ce que l'on transmet, c'est l'information
  - La connaissance est ce que l'on tire de l'information en fonction d'antécédents propres.

- Modèle de la connaissance de Rasmussen

Homme  $\beta$  Connaissances à Système

## Modèle de la connaissance de Rasmussen

[Rasmussen86]



## Neurosciences et IHM

- Modèle des hémisphères du cerveau

Gauche	Droit
Verbal	Non Verbal
Analytique	Synthétique
Rationnel	Non Rationnel
Logique	Intuitif
Local	Global

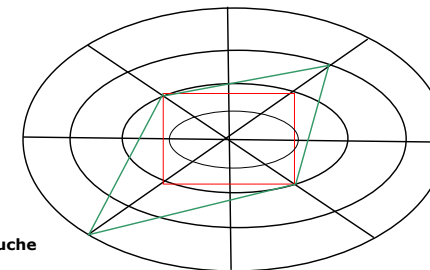
Modèle hémisphérique de Raclé tiré de [Prévo]t

## Neurosciences et IHM

**Cortical gauche**  
 Résolution de pb  
 Mathématique  
 Technique  
 Analyse  
 Logique

**Le raisonnement**

**Cortical droit**  
 Global  
 Artistique  
 Imaginatif  
 Synthétique  
 Conceptuel



**Limnique gauche**  
 Contrôle  
 Planification  
 Organisation  
 Conservateur  
 Administration

**Les émotions**

**Limnique droit**  
 Non Verbal  
 Musicien  
 Spirituel  
 Émotions  
 Contacts humains

Modèle de Hermann tiré de [Prévo]t

## Neurosciences et IHM

Observez cette liste et dites la **COULEUR** de chaque mot, pas le mot lui-même

JAUNE BLEU ORANGE  
 NOIR ROUGE VERT  
 VIOLET JAUNE ROUGE  
 ORANGE VERT NOIR  
 BLEU ROUGE VIOLET  
 VERT BLEU ORANGE

la partie droite de votre cerveau  
 essaye de dire la couleur  
 alors que la partie gauche  
 insiste pour lire le mot

## Neurosciences et IHM

Regardez l'image: vous allez progressivement voir apparaître des visages.

C'est très troublant (il paraît qu'il y en aurait 11)...



Remarquez qu'une fois certains visages repérés, il est impossible de ne plus les voir... Cela illustre notre incapacité à avoir un regard neuf sur un sujet déjà organisé dans le cerveau.

Si vous n'avez aucun problème pour voir l'image sans les visages, il vous est conseillé de vous adresser à un spécialiste de toute urgence!

## Ergonomie et IHM

- L'ergonomie énonce des règles et des propositions de bon sens établies le plus souvent expérimentalement, par observation de l'interaction homme-machine
- Étudie l'adaptation de l'outil à l'homme (et non le contraire)
  - Poste de travail (seul et dans son contexte d'utilisation)
  - Modes d'interaction
  - Respect d'un style

## Les Apports

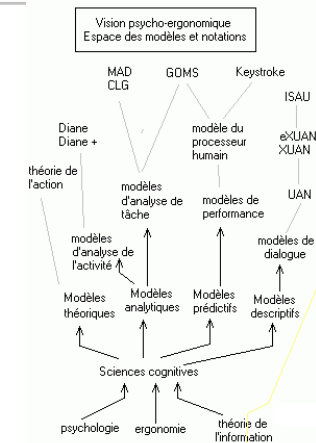


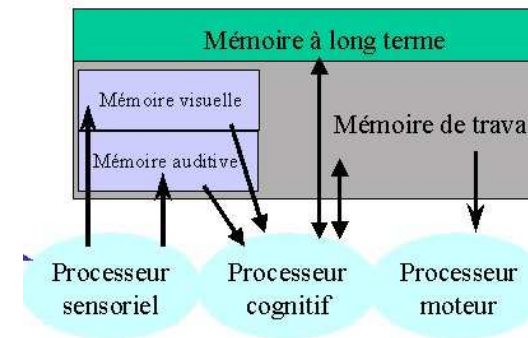
Figure tirée de [Beldame01]

## Les Apports

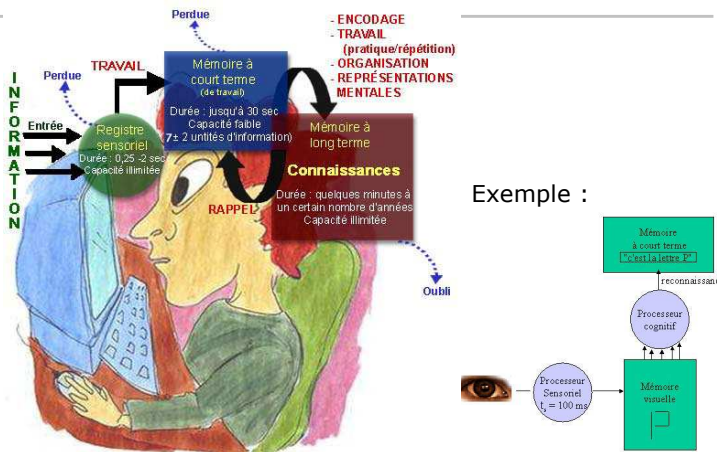
- **de la psychologie cognitive et neurosciences**
  - Un modèle théorique général : le modèle du processeur humain
  - Deux modèles behavioristes : GOMS et Keystroke
  - Un modèle cognitiviste : la théorie de l'action de D. Norman
- **de l'ergonomie**
  - Les 5 règles fondamentales
  - Des stratégies médiatiques

## Modèle du Processeur Humain

[S. Card, T. Moran et A. Newell, 1983]



## Système sensoriel



## Système Sensori-Moteur

- Le système moteur est responsable des mouvements : manipulations des unités physique de commande (clavier, souris, etc)
- Un mouvement est une suite de micromouvements discrets. Chacun s'accomplit en moyenne en  $t_m = 70$  ms
- Loi de Fitts :

$$T = I \log_2 (2D/L)$$

- T = temps pour que la main se place sur la cible
- D = distance de la main à la cible
- L = largeur de la cible
- I = constante évaluée à 0.1s

## Système cognitif : Mémoire à court terme

- Ce sont les "registres" du processeur cognitif
  - Les informations sont comparées par le processeur cognitif au contenu de la mémoire à long terme
  - Le contenu est organisé en mnèmes ("chunks")
  - Nombre "magique" de Miller :  $mc = 7 \pm 2$  mnèmes
  - Problèmes d'interférences : lorsque la mémoire à court terme est saturée, l'arrivée de nouveaux mnèmes efface ceux qui n'ont pas fait l'objet d'une réactivation
- Rem :
- 7 mots français / 7 mots inconnus
  - Phrase vs mots séparés

## Système cognitif : Mémoire à long terme

- Mémoire de masse (taille infinie) pouvant être lue et modifiée
- Elle contient un réseau de mnèmes qui représentent des connaissances procédurales (savoir-faire) et des connaissances déclaratives (savoir)
- Un mnème est transféré de la mémoire à court terme vers la mémoire à long terme par association avec des mnèmes existants
- Traitement de surface (sur la structure du mnème) : "Plage" commence par une majuscule
- Traitement intermédiaire acoustique) : "Plage" rime avec paysage
- Traitement profond (sémantique) : "Plage" me fait penser à un agréable jour de vacances
- Plus le traitement est profond, plus la mémorisation et la récupération (retrieval) sont bonnes (dépend du temps consacré au traitement)

## Techniques pour favoriser la mémorisation à long terme

- Reformuler l'information
  - Ajouter du sens (raconter une histoire)
  - Imagination visuelle (technique des sophistes)
  - Organiser (chunking : créer un mnème)
  - Faire des liens avec des connaissances existantes (particulières ou générales - catégories)
- Rem :
- Nombres de fenêtres chez vous

## Les apports du modèle

- Pour les systèmes d'apprentissage
  - Action (Théorie des 3 A : Auditeur-Acteur-Auteur)
  - Emotion
  - Pédagogie globale & différenciée
- Notion de charge mentale :
  - Mémoire court terme & mémoire de travail
  - Mémoire à long terme (contexte)
  - Surcharge cognitive
  - Fatigue, énervement, stress

## Intérêt et Critiques du modèle

### • Intérêt

- Définit un cadre fédérateur à la diversité des connaissances en psychologie cognitive
- Permet de faire des calculs approximatifs pour évaluer a priori certaines performances (de bas niveau) de l'utilisateur

### • Limites

- Ne donne aucune indication sur la structure des représentations mentales et les opérations de haut niveau (comment se forme un concept ? Comment la mémoire se reconstruit-elle ? Comment les interruptions sont gérées ?...)
- Pas de méthode de conception

## Modèle GOMS

- GOMS [S. Card, T. Moran et A. Newell, 1983] se contente de modéliser le comportement observable de l'utilisateur (approche behavioriste) et ne cherche pas à décrire les états mentaux et les traitements internes (approche cognitiviste)
- GOMS (Goal, Operator, Method, Selection) s'appuie sur le principe de rationalité du MPH
  - *Un but est une structure symbolique qui définit un état recherché. A chaque but est associé un ensemble de méthodes qui permettent de l'atteindre*
  - *Un opérateur est une action élémentaire dont l'exécution provoque un changement d'état (état mental de l'utilisateur et/ou état de l'environnement)*
  - *Une méthode est une suite conditionnelle de buts et d'opérateurs où les conditions font référence au contenu de la mémoire à court terme et à l'état de l'environnement. Une méthode est un "plan d'action" acquis par expérience*
  - *Une règle de sélection exprime le choix d'une méthode lorsque plusieurs méthodes conduisent à un même but*

## Exemple GOMS

- Dans un éditeur pleine page type Emacs, le but de l'utilisateur est de placer le curseur en bas de la page
  - Méthode M1 { prendre la souris ; déplacer le pointeur jusqu'au point désiré ; sélectionner }
  - Méthode M2 { tant que le curseur n'est pas sur la ligne désirée, taper Ctrl-n ; tant que le curseur n'est pas dans la colonne désirée, taper Esc-f }
- Règle de sélection : Si le but à atteindre est de placer le curseur au bas de la page et Si la position actuelle du curseur est loin du bas de la page alors utiliser M1, sinon utiliser M2

## Modèle Keystroke

- Keystroke [S. Card, T. Moran et A. Newell, 1983] est une version simplifiée de GOMS
- Etant donné :
  - Une tâche
  - Un langage de commande (textuel ou graphique)
  - Les paramètres caractéristiques des capacités motrices de l'utilisateur
  - Les paramètres mesurant les temps de réponse du système
  - Une méthode pour réaliser la tâche
- prédire le temps d'exécution de cette tâche par un utilisateur expert
- **Remarque : on suppose que la méthode est unique, on ne prend pas en compte l'opération de choix s'il y a plusieurs méthodes candidates pour le même but**



## Critique des modèles GOMS et Keystroke

- **Avantages :**

- Analyse descendante (top-down) du comportement d'un utilisateur à partir d'un but, ou analyse ascendante (bottom-up) par assemblage de comportements élémentaires : démarches familières à l'informaticien, faciles à mettre en oeuvre
- Modèles fournissant des mesures -> modèles prédictifs

- **Inconvénients :**

- L'utilisateur est supposé ne jamais faire d'erreur (expert)
- Ces modèles ne fournissent aucune aide à la conception, ils ne permettent que d'évaluer, de comparer différentes options de conception d'interfaces
- Les évaluations des temps élémentaires moyens sont imprécises
- Pas de vision globale de l'activité de l'utilisateur : les tâches sont considérées comme indépendantes
- Modèles purement behavioristes : on ne cherche pas à comprendre les représentations internes de l'utilisateur

## La théorie de l'action de Norman [Norman86]

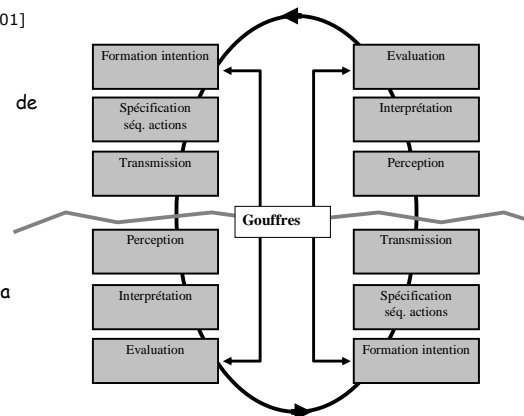
- **Approche cognitive de la modélisation de l'utilisateur :**

- l'individu élabore un modèle conceptuel du système informatique.
  - Le comportement est conditionné par l'environnement
- ET**
- par la représentation interne que l'utilisateur se fait du système

## Théorie de l'action & Ingénierie cognitive

Tiré de [Tarpin01]

Univers  
L'utilisateur



Univers de la  
machine

## La théorie de l'action de Norman [Norman86]

- **Deux types de modèles :**

- Le modèle de conception, sur lequel s'est appuyé la réalisation du système, caractérisé par un ensemble de variables "physiques" et leurs relations
- Le modèle de l'utilisateur : représentation mentale que l'utilisateur se fait de sa tâche et de l'outil caractérisée par un ensemble de variables "psychologiques"
- Une erreur est commise :
  - Soit parce que l'interface n'explique pas l'état effectif du système
  - Soit parce que la correspondance entre les variables physiques et les variables psychologiques est incohérente ou incomplète ou trop compliquée
  - Soit parce que les dispositifs de contrôle des variables physiques ne sont pas adaptés à la tâche

## Théorie de l'action & Ingénierie cognitive

- Une Interface doit être conçue dans le but de réaliser une activité précise, les questions sont :
- Quel est le but ? Pourquoi ? Activité/Tache
- Quels sont les objets, actions ? Quoi ? Actions
- Quel sont les méthodes ? Comment ? Opérations

## Apport de la théorie de l'action

- La réalisation d'une **tâche** par l'utilisateur met en jeu 7 étapes :
  - 1) l'établissement d'un but
  - 2) la formation d'une intention
  - 3) le spécification d'une suite d'actions
  - 4) l'exécution des actions
  - 5) la perception de l'état du système
  - 6) l'interprétation de l'état du système
  - 7) l'évaluation de l'état du système par rapport au but
- Elle permet d'expliquer les réussites, difficultés et erreurs de l'utilisateur
- Elle permet de classer les besoins de l'utilisateur et de mettre un peu d'ordre dans l'ensemble des règles ergonomiques trouvées expérimentalement

## Evaluation des Interfaces



## Évaluation d'IHM

- Toujours les 2 critères :
  - Utilisabilité :
    - Qualité de l'interaction : facilité d'usage et d'apprentissage, documentation
  - Utilité :
    - Capacité fonctionnelle
    - performance

## Méthodes d'évaluation d'IHM

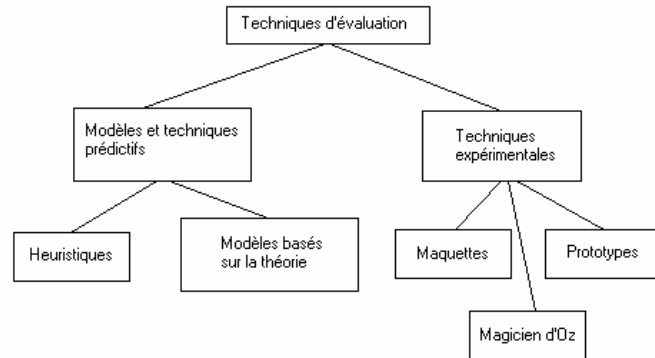
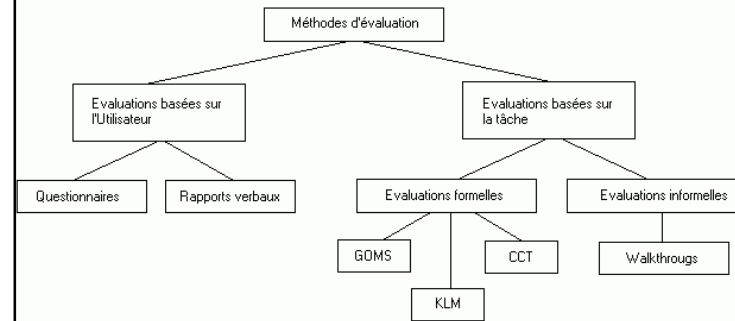


Figure tirée de [Beldame01]

## Méthodes d'évaluation d'IHM



Classification de Karat.  
Figure tirée de [Beldame01]

## Méthodes d'évaluation d'IHM

- Sans utilisateurs :
  - Ballade cognitive (cognitive walkthrough)  
[Lewis, Polson, Wharton, Rieman, 1992]
  - Analyse analytique : basée sur les modèles (GOMS, Keystroke)
  - Évaluation heuristique [Nielsen, Molich 1990]
- Avec utilisateurs :
  - Test d'utilisation : expérimentation, monitoring, ...
  - Interviews

## Évaluation de sites Web

- Reconnaître plutôt que se rappeler :
  - pas trop d'informations dans une page (surtout dans les premières)
  - les liens sur les images réactives doivent être faciles à trouver
- Flexibilité et efficacité :
  - rendre vos pages faciles à marquer
  - Esthétique et minimalisme
  - pas de décoration qui distraie
  - avoir plusieurs niveaux de détails et pouvoir zoomer dans les deux sens

## Évaluation de sites Web – Quelques règles

- L'état du système doit être visible
  - Deux préoccupations permanentes pour l'utilisateur : où suis-je ? et où puis-je aller?
- Le système doit être le reflet du monde réel
  - Parler le langage de l'utilisateur et suivre les conventions du domaine
- L'utilisateur doit garder le contrôle et être libre
- Ne pas imposer à l'utilisateur des contraintes liées à la technique
- Être consistant et respecter les standards
- les labels des liens et boutons doivent être identiques aux titres des pages auxquels ils renvoient. Votre site n'est pas une île au milieu du Web !
- consistence des couleurs (fonds) et des formes
- Prévenir les erreurs : faire un pré-traitement sur la station cliente pour corriger les erreurs grossières

## Évaluation de sites Web

- Aider l'utilisateur à reconnaître, diagnostiquer et réparer les erreurs
  - Tout message d'erreur doit proposer une solution
- Aide et documentation
  - Ne pas se contenter d'ajouter des pages d'aide, mais intégrer la documentation complète dans le site et créer des liens bidirectionnels entre les pages du site et les pages de documentation

## Exemples

- <http://ictt.insa-lyon.fr>
- <http://www.insa-lyon.fr>
- <http://www.sncf.fr>
- <http://www.evian.fr>

## Normes et standards

- Qu'est-ce qu'une norme ?
  - Les normes sont des accords documentés contenant des spécifications techniques ou autres critères précis destinés à être utilisés systématiquement en tant que règles, lignes directrices ou définitions de caractéristiques pour assurer que des matériaux, produits, processus et services sont aptes à leur emploi
- Qui produit les normes ?
  - Organismes internationaux : ISO , IEC et ITU-T
  - Organismes européens : CEN/CENELEC et ETSI
  - Organismes nationaux : AFNOR
  - Entreprises, associations, consortiums

## Normes ISO pour les IHM (TC 159 / SC 4)

- Exigences ergonomiques pour travail de bureau avec terminaux à écrans de visualisation :
  - Introduction générale ISO 9241-1:1997
  - Guide général concernant les exigences de tâches ISO 9241-2:1992
  - Exigences relatives aux écrans de visualisation ISO 9241-3:1992
  - Exigences relatives aux claviers ISO 9241-4:1998
  - Exigences relatives à l'aménagement du poste de travail et aux postures ISO 9241-5
  - Guide général relatif à l'environnement de travail ISO 9241-6
  - Exigences d'affichage concernant les réflexions ISO 9241-7:1998
  - Exigences relatives aux couleurs affichées ISO 9241-8:1997
  - Exigences relatives aux dispositifs d'entrée autres que les claviers ISO 9241-9
  - Principes de dialogue ISO 9241-10:1996
  - Lignes directrices relatives à l'utilisabilité ISO 9241-11:1998
  - Présentation de l'information ISO 9241-12

## Normes ISO pour les IHM (suite)

- Exigences ergonomiques pour travail de bureau avec terminaux à écrans de visualisation
  - Guidage de l'utilisateur ISO 9241-13:1998
  - Dialogues de type Menu 9241-14:1997
  - Dialogues de type langage de commandes 9241-15:1997
  - Dialogues de type manipulation directe 9241-16
  - Dialogues de type remplissage de formulaires 9241-17:1998
- Processus de conception centrés sur l'individu pour les systèmes interactifs ISO/DIS 13407

## Pour conclure : esthétique et technique

- Une IHM doit aussi être belle :
  - il faut satisfaire autant les besoins spirituels que les besoins matériels
  - Allier économie structurelle et perfection fonctionnelle
  - La formule "tout ce qui est fonctionnel est beau" n'est qu'à moitié vraie.

## FIN DU COURS