



---

# **Cours IHM-1**

## ***Conception des interfaces***

### **1 - Introduction**

### **Concepts généraux**

---

# Interface Homme-Machine [1]

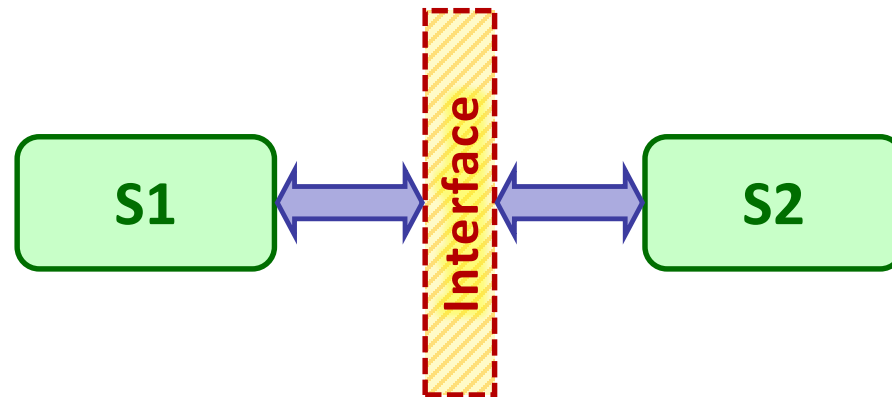


- L'étude de l'**Interface Homme-Machine (IHM)** appelée également **Interaction Homme-Machine** (*Human Computer Interaction* ou **HCI**) s'occupe de tous les moyens et outils permettant à un humain de contrôler, de communiquer et d'interagir avec un système interactif (une machine, au sens large).
- C'est un domaine scientifique qui est au confluent de **nombreuses disciplines** qui concernent :
  - L'humain et ses **aspects cognitifs**
    - ⇒ Sciences cognitives, ergonomie physiologique et cognitive, utilisabilité
    - ⇒ Perception, physiologie, psychologie, sociologie, ...
    - ⇒ Modélisation du comportement, modèles mentaux, ...
  - La machine et ses **aspects techniques**
    - ⇒ Utilisation et codage de composants d'interaction (*widgets*)
    - ⇒ Capture et traitement des actions de l'utilisateur (événements)
    - ⇒ Architecture logicielle (découplage entre les modules)

# Interface Homme-Machine [2]



- Au sens large, la notion d'**interface** représente un dispositif placé à la limite commune de deux entités/systèmes qui communiquent.



- Dans le cas de l'**IHM**, on va trouver d'un côté le système interactif (la machine **M**) et de l'autre l'utilisateur (l'humain **H**).
- L'interface (**I**) se charge dans ce cas de mettre en communication :
  - La représentation externe de l'état de la machine (image du système)
  - et*
  - Les organes sensoriels (perception) et moteurs (actions) de l'utilisateur

# Interface Homme-Machine [3]



- Il existe de nombreuses définitions de l'interface ou l'interaction homme-machine et le domaine d'étude est très vaste.
- Une autre définition de l'interaction homme-machine met en évidence trois notions importantes :
  - *L'ensemble des aspects de la **conception**, de l'**implémentation** et de l'**évaluation** des systèmes interactifs*
    - ⇒ Conception : Aspects créatifs (*design*)
    - ⇒ Implémentation : Aspects techniques de réalisation (*programmation*)
    - ⇒ Évaluation : Aspects liés à la validation de l'ergonomie (*tests*)
- Et deux autres définitions plus formelles :
  - *L'interaction homme-machine consiste à mettre en œuvre les moyens et les techniques permettant de franchir la discontinuité entre les référentiels de deux agents.*
  - *L'art de concevoir et de réaliser des représentations pertinentes.*

# Éléments de l'IHM [1]

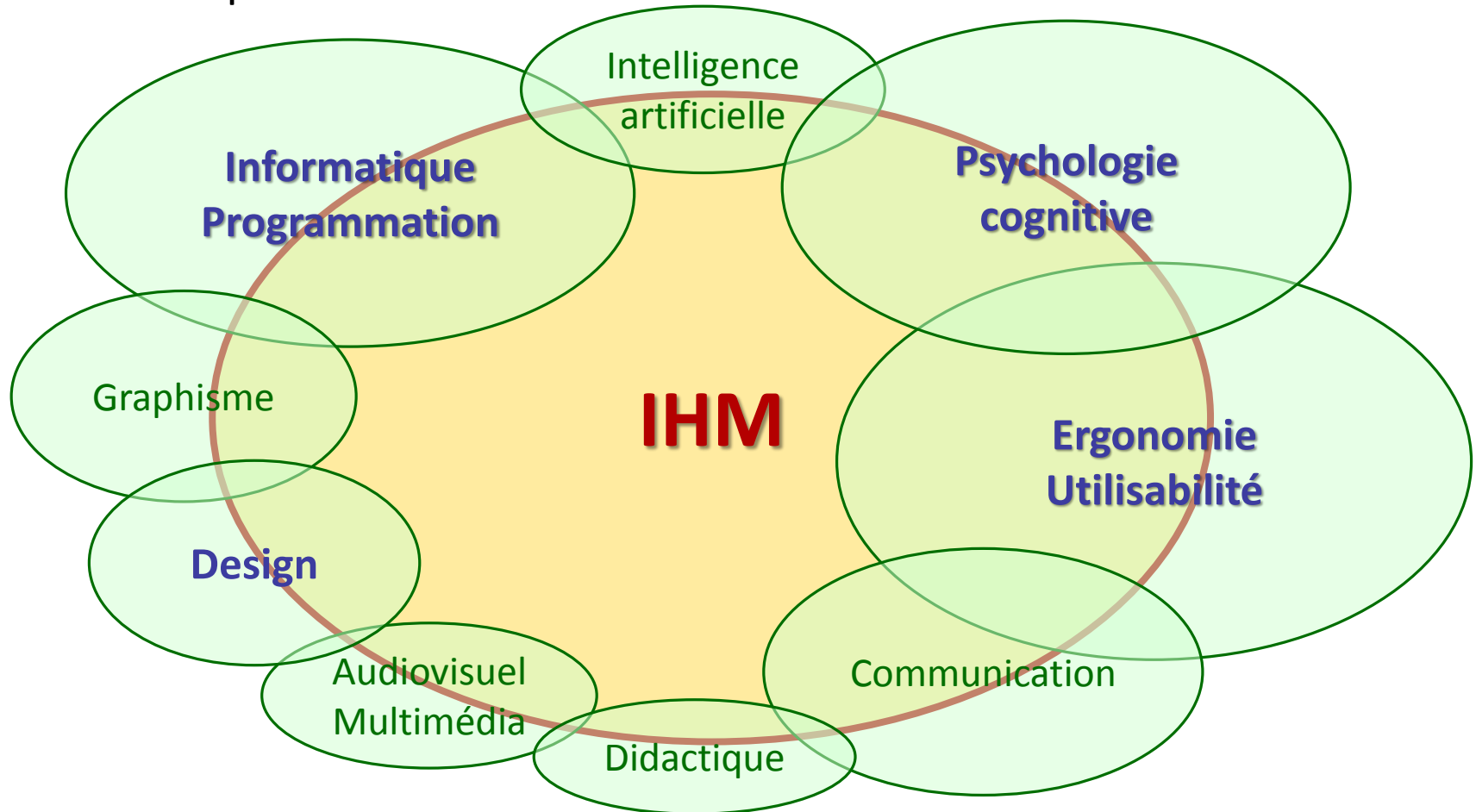


- Dans l'étude des interfaces homme-machine il y a donc trois éléments à considérer : l'**utilisateur** (l'homme), le système interactif (la machine) et leur **manière de communiquer** (l'interface).
- Une bonne connaissance des ces trois domaines est donc nécessaire à la réalisation d'une interface homme-machine réussie.
- En plus de ces trois éléments, il faut naturellement avoir une très bonne **connaissance du domaine** qu'est censé couvrir le logiciel, le site web, etc. (que l'on nomme parfois l'**espace du problème**).
- Le développement des interfaces utilisateurs nécessite de la part de l'informaticien de vastes compétences pluridisciplinaires.
- En plus de compétences informatiques, il doit posséder un certain nombre de *soft-skills*, des qualités humaines et relationnelles telles que qualité d'écoute, communication, créativité, sens de l'initiative, empathie, collaboration, etc.

# Éléments de l'IHM [2]



- L'interaction homme-machine : un domaine fortement pluri- et interdisciplinaire.



# Approches de l'IHM [1]

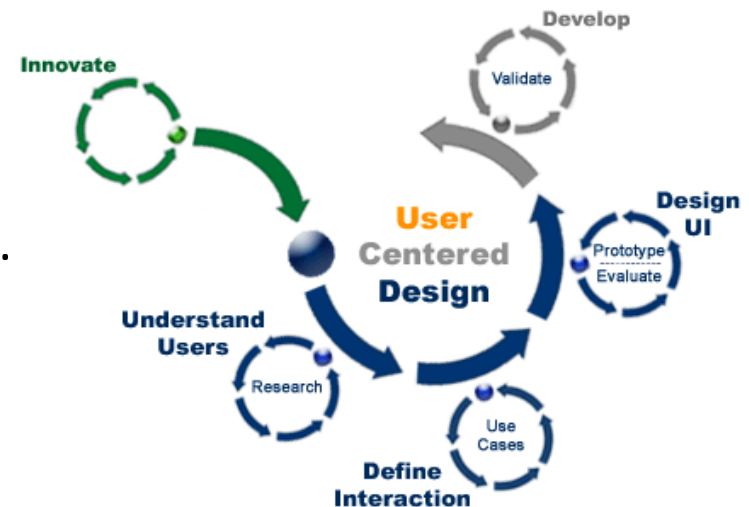


## ■ Approche *technocentrée* :

- Développement **centré sur la machine** et ses possibilités techniques
- L'utilisateur doit s'adapter à la machine
- Une grande tentation pour les développeurs
  - ⇒ Imposer des technologies qui leur plaisent
  - ⇒ Privilégier la création de systèmes efficaces sur le plan technique

## ■ Approche *anthropocentrée* :

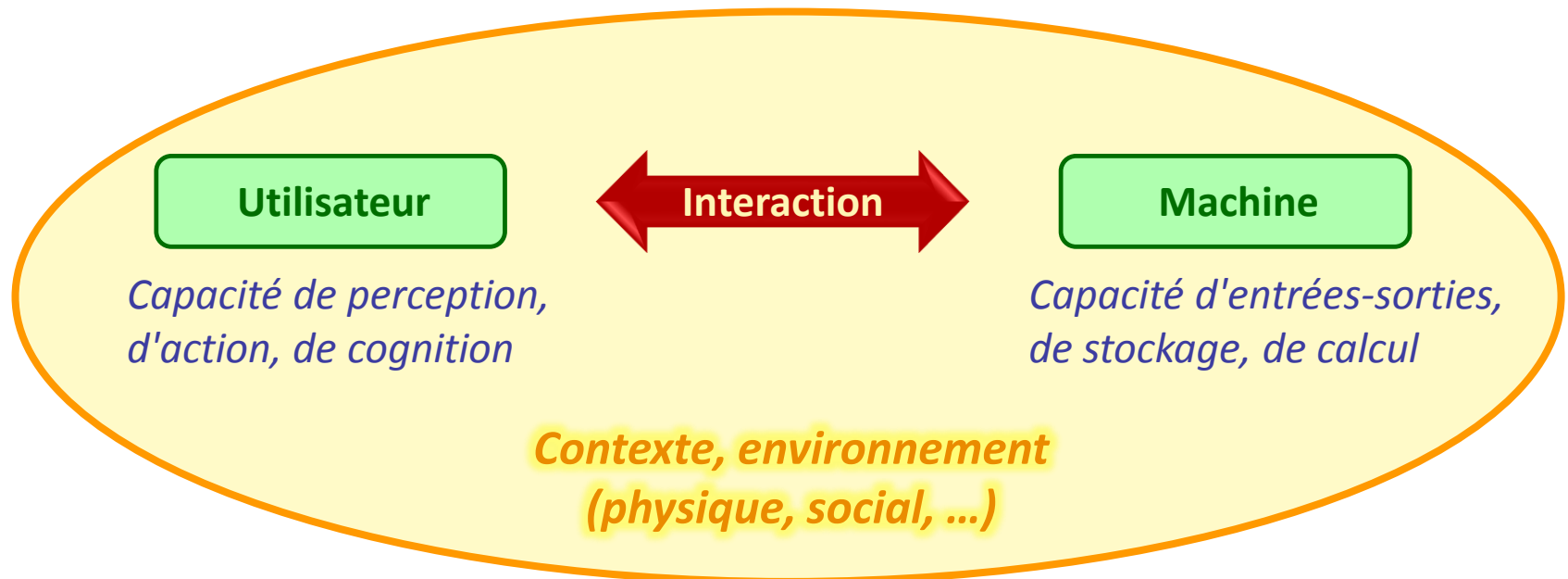
- Développement du logiciel **centré sur l'utilisateur** et ses besoins
- La machine doit s'adapter à l'utilisateur
- Nécessite que les développeurs connaissent les besoins des utilisateurs, le contexte d'utilisation, etc.
- Approche à privilégier, connue sous le nom *User Centered Design*



# Approches de l'IHM [2]



- La **conception centrée sur les utilisateurs** (UCD) nécessite, de la part du développeur, de bien connaître :
  - Les **caractéristiques d'un humain** (perception, comportement, ...)
  - Les **techniques de programmation** (*widgets*, événements, algorithmes, ...)
  - Le **problème à résoudre** (les tâches de l'utilisateur) ainsi que le **contexte d'utilisation** de l'application





# Aspects subjectifs

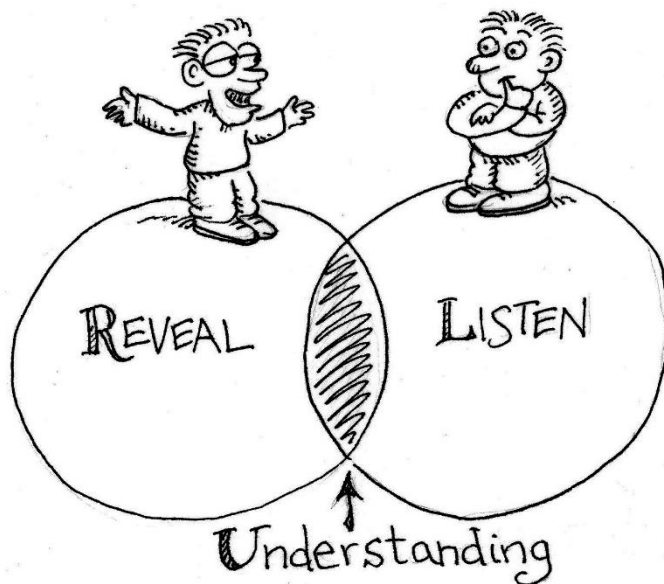


- Le fait qu'une partie du domaine de l'IHM concerne des utilisateurs souvent divers et variés, implique que la conception peut faire intervenir une part de subjectivité.
- Certaines **composantes artistiques/esthétiques** (la disposition des composants, le choix des couleurs, les sons, les images, etc.) peuvent être appréciées de différentes manières selon les goûts et les préférences de chaque personne.
- Il a cependant été démontré que la qualité de l'interaction avec un système (ce que l'on nomme l'*expérience utilisateur*) est régie par un certain nombre de règles qui sont communément admises, validées par des tests et qui sont indépendantes des aspects subjectifs.
- Les aspects subjectifs restent donc marginaux et les éléments qui concernent les préférences individuelles peuvent être réglés en offrant à l'utilisateur des options pour configurer et personnaliser son interface (critère ergonomique d'adaptabilité).

# Communication / Dialogue [1]



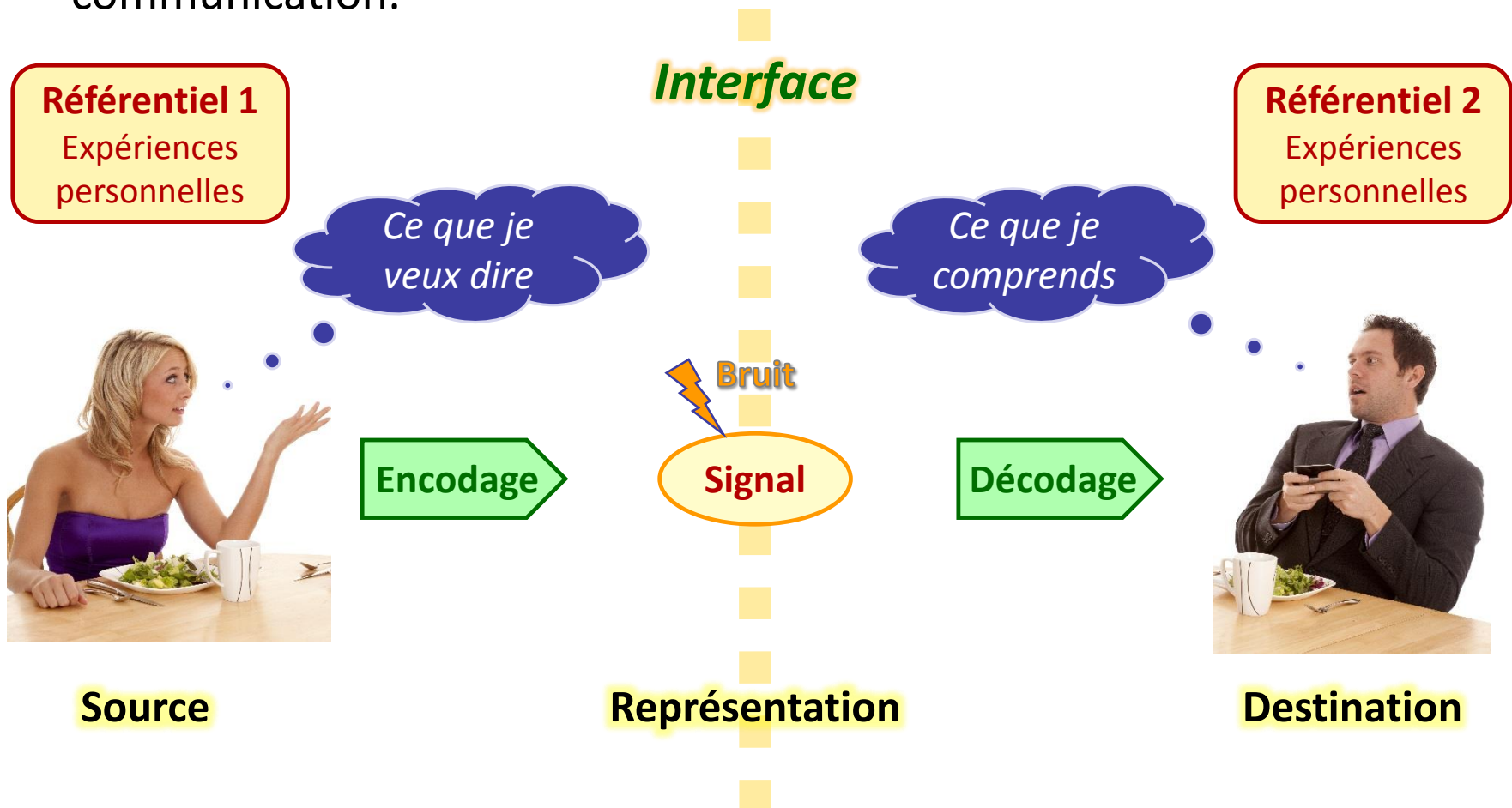
- Concevoir une bonne interface consiste essentiellement à résoudre un **problème de communication**.
- Déjà entre humains ce n'est pas toujours facile...



# Communication / Dialogue [2]



- Il est important de comprendre les mécanismes fondamentaux de la communication.



# Communication / Dialogue [3]



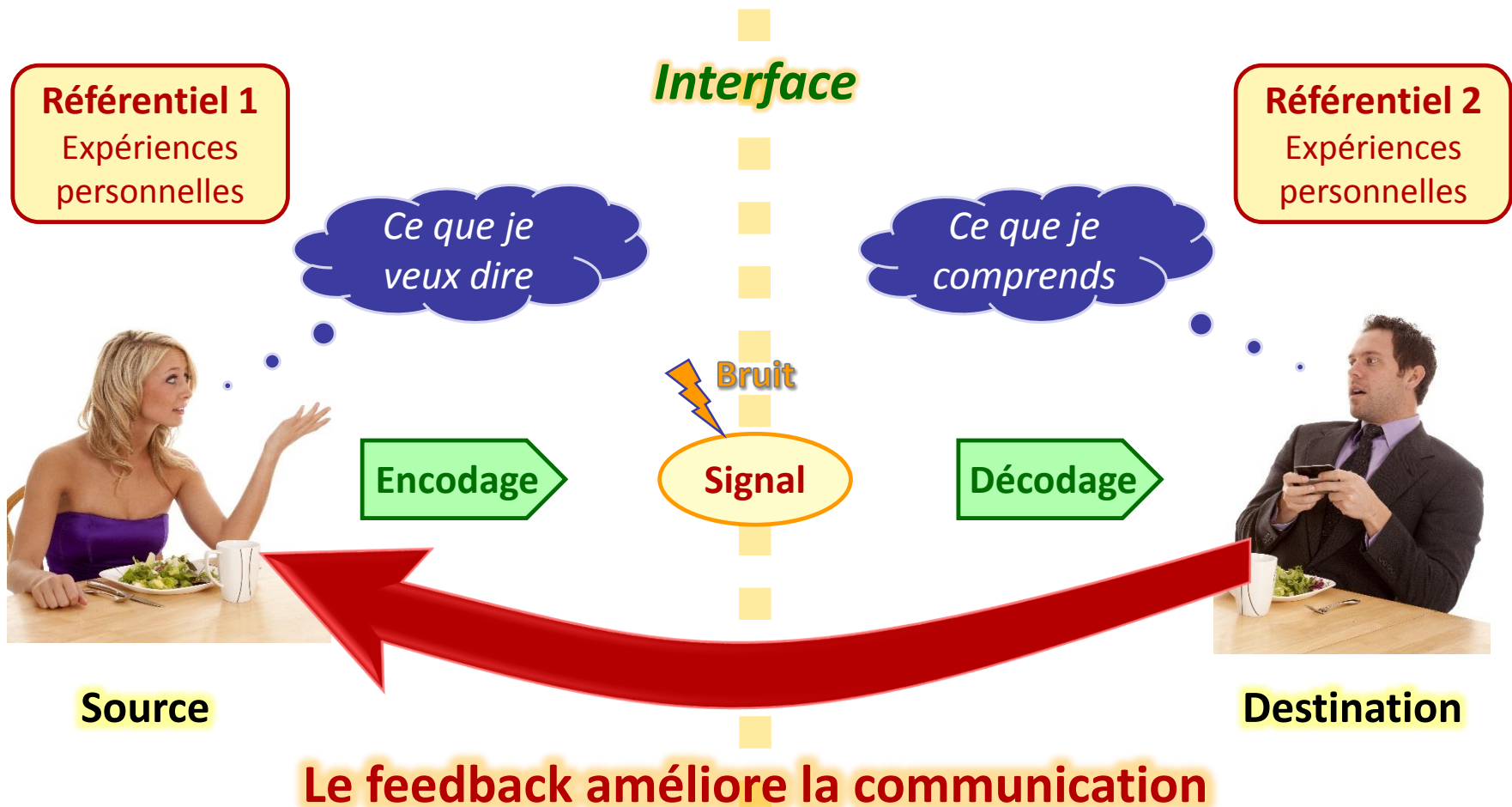
- La représentation des signaux échangés peut prendre des formes très diverses :
  - Textes
  - Symboles
  - Images
  - Voix, sons
  - Gestes
  - Expressions faciales (sourire, étonnement, ...)
  - Attitudes corporelles (langage du corps)
  - Modes haptiques/tactiles (pression, vibration, température, ...)
  - Modes olfactifs (parfum, ...)
  - ...

*D'une manière générale, les éléments échangés peuvent être de n'importe quel type.*

# Communication / Dialogue [4]



- Naturellement la communication est bidirectionnelle (dialogue).



# Communication / Dialogue [5]



- Dialoguer avec une machine n'est généralement pas plus simple...

© 2000 Randy Glasbergen.  
www.glasbergen.com



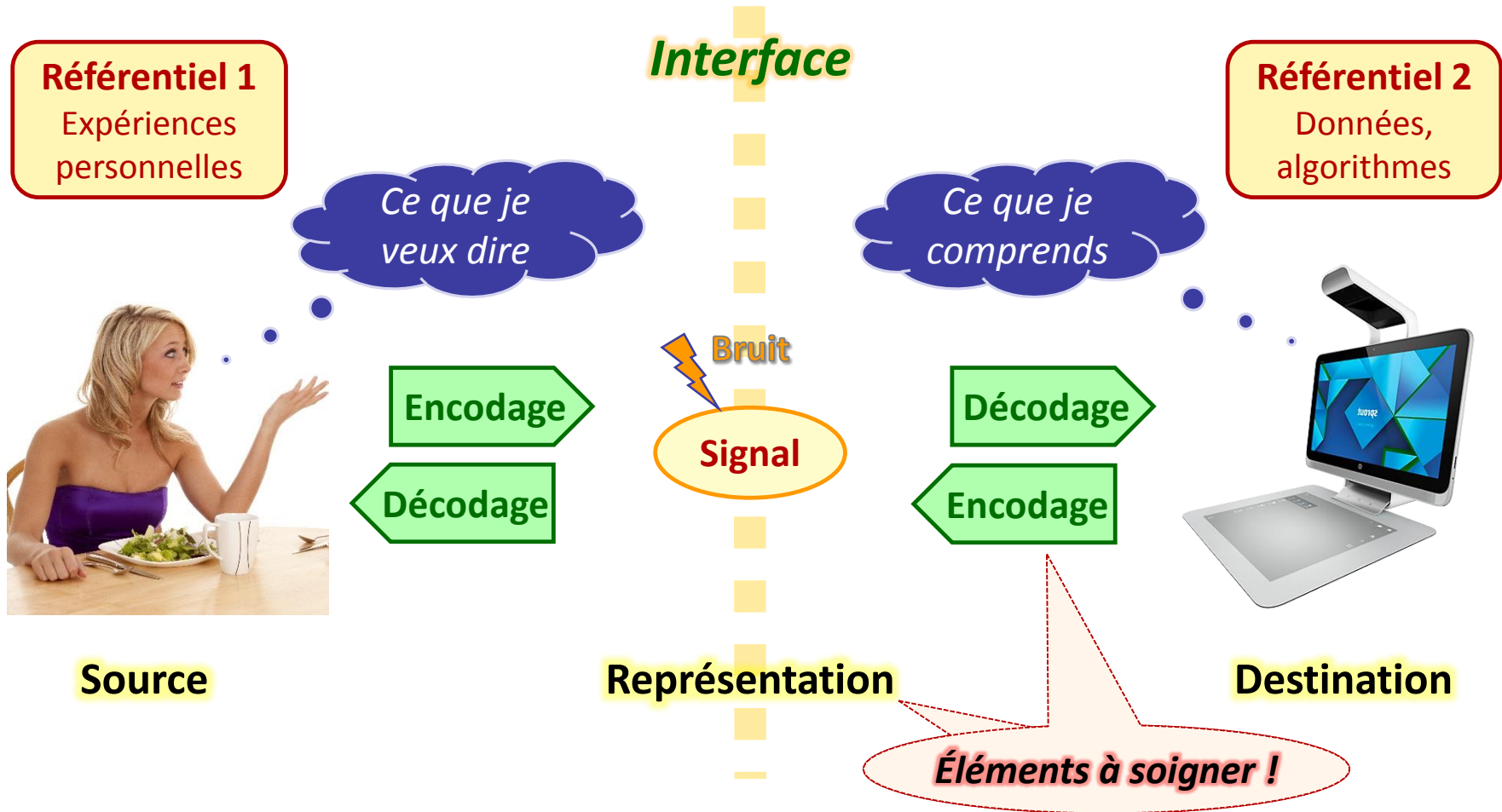
"THE COMPUTER SAYS I NEED TO UPGRADE MY BRAIN  
TO BE COMPATIBLE WITH ITS NEW SOFTWARE."



# Communication / Dialogue [6]



- Dans l'idéal, l'interface utilisateur de la machine doit s'adapter à la manière de communiquer de l'humain (et non l'inverse).



# L'humain / L'utilisateur [1]



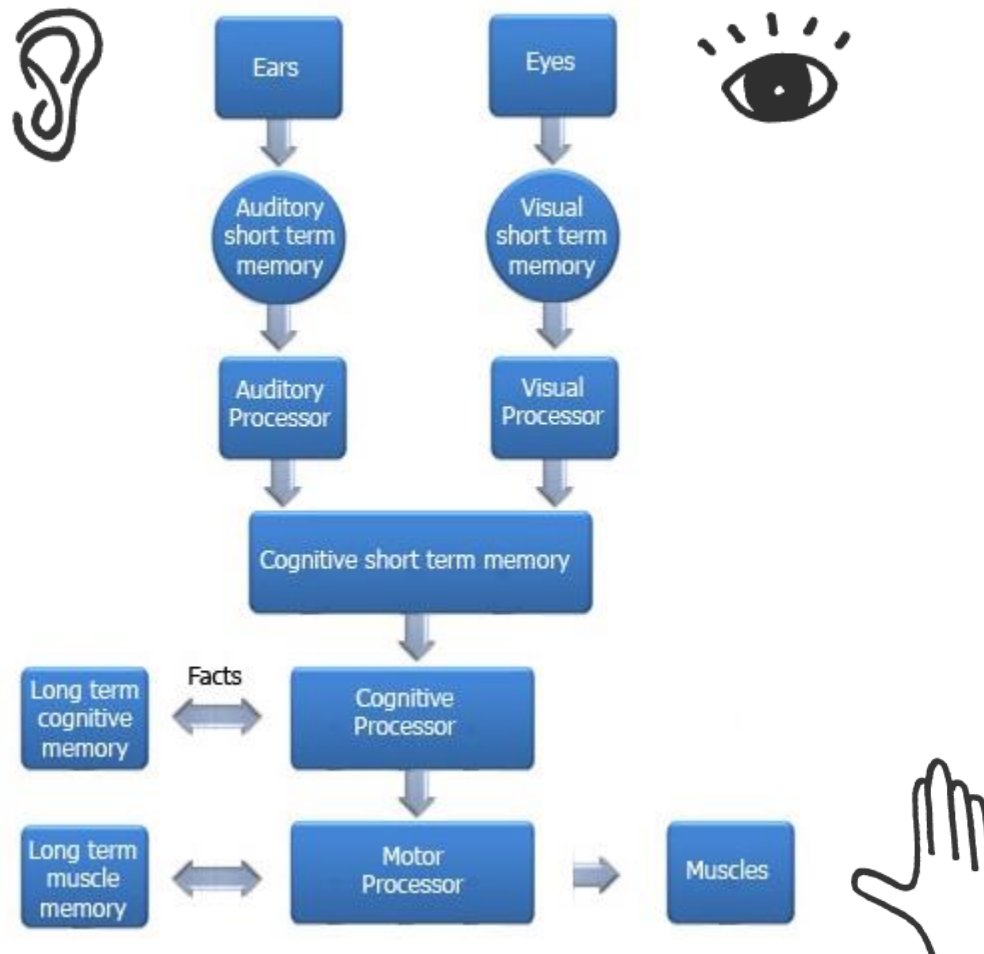
- C'est l'élément essentiel (prioritaire) du trio interface homme-machine (la machine est au service de l'humain et non le contraire).
  - Quels sont ses besoins, ses attentes (phase d'analyse) ?
  - Quelles sont ses capacités physiques, ses limitations (handicaps) ?
  - Quelles sont ses capacités cognitives, ses connaissances du domaine traité, le contexte dans lequel il évolue ?
- Connaître certains éléments du système sensoriel et du comportement humain est important pour concevoir une bonne interface. Par exemple :
  - Qu'il s'attend à avoir une rétroaction (*feedback*) pour chacune de ses actions (comme lorsqu'il manipule des objets du monde réel).
  - Qu'il est distrait par les plus petits mouvements à la périphérie de son système visuel (même s'il ne voit bien qu'au centre).
  - Que son cerveau interprète (parfois de manière irrépressible) les informations provenant de ses sens (et qu'il est facile de le tromper).



# L'humain / L'utilisateur [2]



- Modéliser un humain est le sujet de nombreuses recherches.



# La machine / Le système interactif



- Pour un humain, c'est une **boîte noire** :
  - Pas de feedback implicite (il faut le programmer explicitement).
  - Les informations stockées et les opérations effectuées ne sont pas directement visibles (opacité).
- Son mode de fonctionnement est assez différent du nôtre :
  - **[+]** Peut mémoriser un grand nombre d'informations et les retrouver rapidement et sans pertes
  - **[+]** Peut exécuter des instructions autant de fois que nécessaire sans erreurs (pas de notion de fatigue)
  - **[-]** Incapable de faire certaines choses qui nous paraissent triviales (tirer des conclusions, laisser tomber certains éléments négligeables au profit d'éléments plus importants, comprendre le langage naturel même des phrases simples, corréler des informations, ne pas répéter les mêmes erreurs, etc.)
  - **[-]** L'intelligence artificielle reste limitée pour l'instant (même si on constate certains progrès)

# L'interface / Le dialogue [1]



- Connaissant les différences (importantes) entre l'homme et la machine, la question devient : comment trouver un **langage commun** pour que ces deux entités communiquent au mieux ?
- C'est le difficile problème que doit régler l'**interface utilisateur**.
- L'interface utilisateur doit ainsi jouer un rôle d'interprète et traduire le langage de l'utilisateur d'une part et celui de la machine d'autre part pour les mettre en communication (comme le ferait un interprète entre deux personnes ne parlant pas la même langue).
- C'est tout l'art du concepteur de l'interface d'établir le lien entre ce que l'utilisateur souhaiterait (dans l'idéal) et ce que les techniques informatiques peuvent offrir.
- C'est souvent un art du compromis et il faut jongler entre les aspects techniques, les budgets alloués, les délais à respecter, les ressources à disposition, etc.

# L'interface / Le dialogue [2]



- L'interface utilisateur doit mettre en relation les entrées/sorties de la machine avec celles de l'humain.
- Entrées/sorties de la **machine** :
  - Périphériques d'entrée
    - ⇒ Clavier, souris, *touchpad*, écran tactile, joystick, microphone, caméra, ...
  - Périphériques de sortie
    - ⇒ Écran, imprimante, haut-parleurs, écouteurs, retour de force, ...
- Entrées/sorties de l'**humain** :
  - En entrée
    - ⇒ Cinq sens : vue, ouïe, toucher, goût, odorat
  - En sortie
    - ⇒ Système moteur (gestes, mouvements, expressions faciales, regard, ...)
    - ⇒ Voix
- Dans les applications classiques qu'est-ce qui est réellement utilisé ?

# Interfaces graphiques (GUI)



- Dans la majorité des applications classiques, l'interaction homme-machine fait appel aux périphériques de base qui sont : l'**écran**, le **clavier** et la **souris** (et ses substituts).
- Les interfaces utilisateur graphiques (*Graphical User Interface* ou *GUI*) se sont généralisées. Elles utilisent des **composants graphiques** appelés **Widgets** ou **Controls** en combinaison avec un dispositif de pointage et d'action (une souris par exemple).
- La **palette de composants** graphiques varie selon les systèmes, mais on retrouve presque partout des *Widgets* devenus classiques :
  - Boutons (cases à cocher, boutons radio, boutons à deux états, ...)
  - Menus (déroulants, contextuels, méga-menus, ...)
  - Liste à choix (listes simples, listes déroulantes, *spinner*, ...)
  - Champs de saisie (champs texte, zones de texte, éditeur, ...)
  - Barres de défilement, panneaux *scrollables*, ...
  - ...

# Styles d'interaction



- Les **styles d'interaction** (ou **paradigmes d'interaction**) caractérisent différentes manières de dialoguer entre un humain et une machine.
- Quelques styles d'interaction classiques :
  - Style conversationnel (langage de commandes, *shell-scripts*, ...)
  - Interaction par formulaires (*form fill-in*) ou tableurs
  - Sélection par menus (plus nécessaire de se souvenir des commandes)
  - Navigation hypertextuelle (*point-and-click*)
  - WIMP (en réalité un ensemble de styles d'interaction)
  - Manipulation directe (édition WYSIWYG, interaction iconique)
  - Langage naturel écrit ou parlé (*query language*, commande vocale)
  - Interaction gestuelle (*Kinect*, *Leap-Motion*, ...)
  - Interfaces multimodales (qui combinent les modalités)
  - ...

# Manipulation directe [1]



- Parmi les **styles d'interaction** classiques figure la **manipulation directe** qui est utilisée dans un grand nombre d'interfaces.
- Ce style d'interaction (défini par Ben Schneiderman en 1983) donne à l'utilisateur le sentiment d'un engagement direct dans l'action qu'il effectue.
- Par exemple, sélectionner un document et le tirer (*drag&drop*) dans la poubelle est un exemple typique d'une manipulation directe.
- L'interaction directe est particulièrement bien adaptée aux interfaces avec écrans tactiles (smartphones, tablettes, bornes interactives, ...)
- Un slogan (Schneiderman) :
  - *"Point and click instead of remember and type"*



# Manipulation directe [2]



- La manipulation directe est caractérisée par trois principes directeurs :
  - **Représentation permanente** des objets (généralement des icônes qui ont un sens pour l'utilisateur)
  - Utilisation d'**actions physiques** pour effectuer les opérations (soit directement sur l'interface ou via un dispositif de pointage comme la souris)
  - Opérations incrémentales, réversibles et à **effet visible**
- L'inconvénient principal de ce style d'interaction est qu'il ne permet pas facilement l'automatisation des actions (*scripting*).
- On parle parfois de **Manipulation indirecte** lorsque les actions sont déclenchées en passant par des menus et des boîtes de dialogue.
  - Effacer un fichier en passant par le menu contextuel
  - Redimensionner une image en ouvrant une boîte de dialogue



# Interfaces WIMP



- Les interfaces utilisateur qui intègrent ces éléments graphiques (composants visuels / *Widgets* / *Controls*) sont parfois désignées par l'acronyme **WIMP** qui est l'abréviation de :
  - **Window** : Notion de "fenêtre" (zone d'interaction indépendante) (c'est une notion importante même si elle paraît évidente)
  - **Icon** : Éléments graphiques statiques ou éléments d'interaction (images, boutons, champs de texte, bulles d'aide, etc.)
  - **Menu** : Choix d'actions parmi une liste proposée (barres de menu, menus déroulants, menus contextuels, rubans, etc.)
  - **Pointer** : Curseur/Pointeur manipulé par un dispositif de pointage et qui permet d'interagir avec les composants (pointage, sélection, tracé, *drag & drop*)
- Une interface de type WIMP peut faire appel à différents styles d'interaction (même si on parle parfois du style d'interaction WIMP).

# Importance de l'IHM [1]

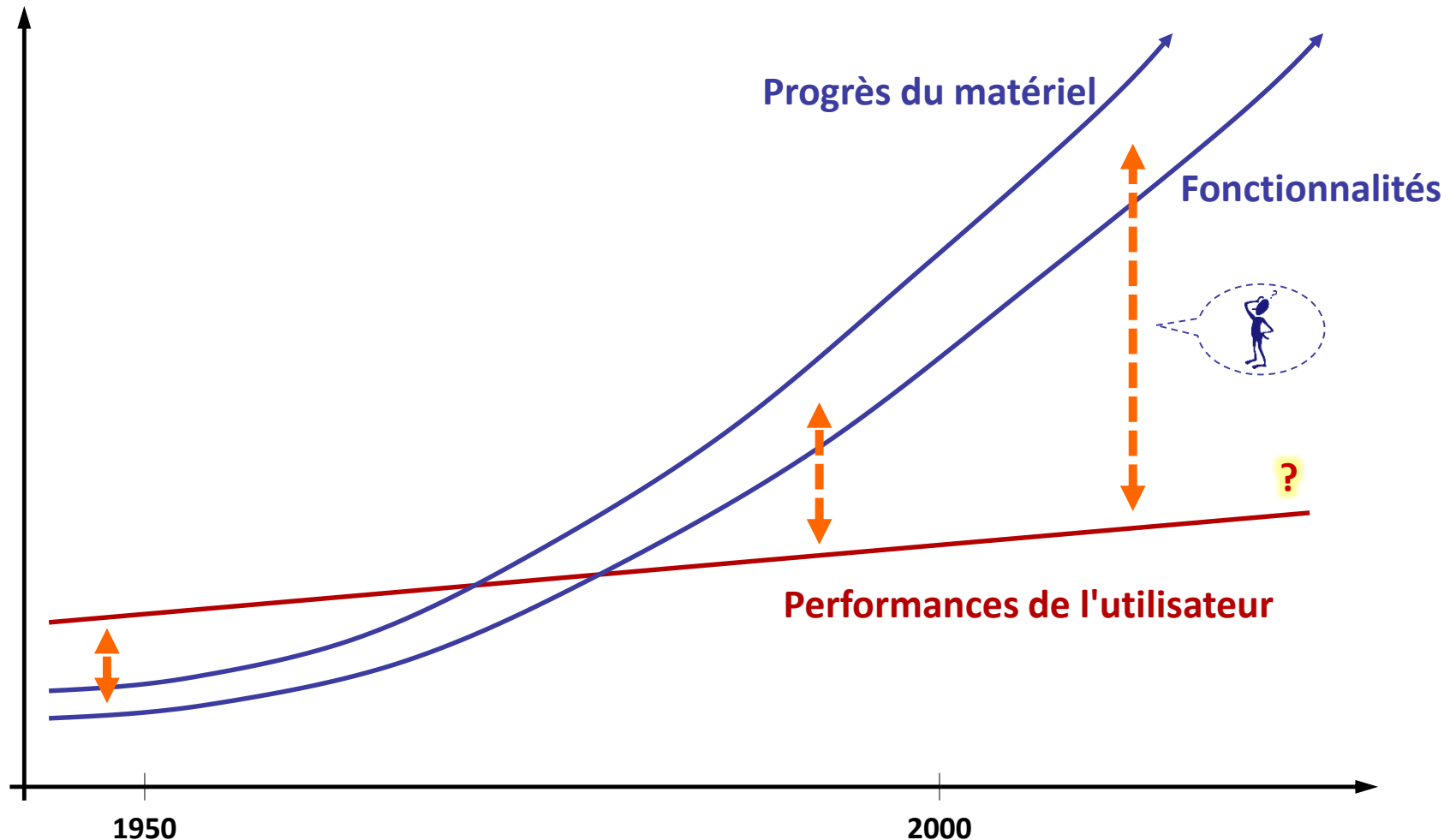


- Créer une bonne interface utilisateur est une **tâche difficile mais extrêmement importante** car c'est souvent sur son l'interface utilisateur qu'un système interactif est jugé.
  - L'interface utilisateur constitue la **vitrine** du produit
- La qualité de l'interface utilisateur peut **conditionner la réussite ou l'échec commercial** d'un logiciel, d'un site web, ... (indépendamment de la qualité intrinsèque de ses fonctionnalités et de son prix).
- C'est également un **facteur de fidélisation** : l'utilisateur achète les yeux fermés lorsqu'il est sûr de pouvoir utiliser facilement le logiciel.
- Ces aspects prennent une importance croissante auprès des décideurs qui ont pris conscience de l'importance de soigner la qualité de l'interface utilisateur.
  - Ces compétences sont de plus en plus recherchées par les employeurs et les responsables RH.

# Importance de l'IHM [2]



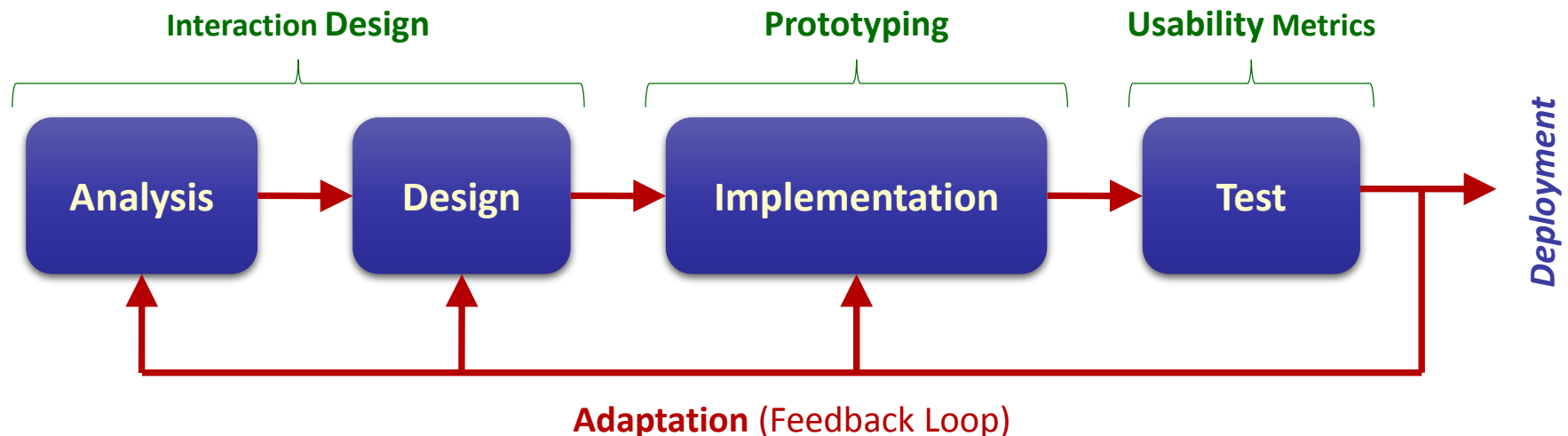
- Évolution de l'interactivité (certains écarts se creusent)



# Cycle de développement



- La clé du succès du développement d'applications comportant des interfaces utilisateur réside dans la mise en place d'un **processus itératif** avec la création de **maquettes (prototypes)** qui seront soumises à l'appréciation des utilisateurs (panel représentatif) puis corrigées/adaptées en fonction des résultats de ces tests.
  - Il est illusoire d'obtenir un bon résultat avec un développement linéaire (développement en cascade)
  - Des itérations sont indispensables (développement en spirale)



# Systèmes interactifs [1]



- La notion de **système interactif** est très vaste et englobe une très grande variété de produits. Dans l'absolu, il peut s'agir de tout dispositif/outil avec lequel un humain interagit.
- Dans ce cours, nous considérerons principalement les **dispositifs comportant du logiciel** et notamment :
  - Applications autonomes (*standalone*)
  - Applications *n*-tiers (client-serveur, en réseau, dans le cloud, ...)
  - Applications mobiles (smartphones, tablettes, ...)
  - Sites et applications web
  - Logiciel embarquée (automates, bornes, ...)
- Certains systèmes plus spécifiques ne seront pas abordés :
  - Robotique, informatique vestimentaire (*wearable*)
  - Réalité virtuelle et augmentée
  - Interfaces multimodales
  - Intelligence artificielle, *machine learning*, ...

# Systèmes interactifs [2]

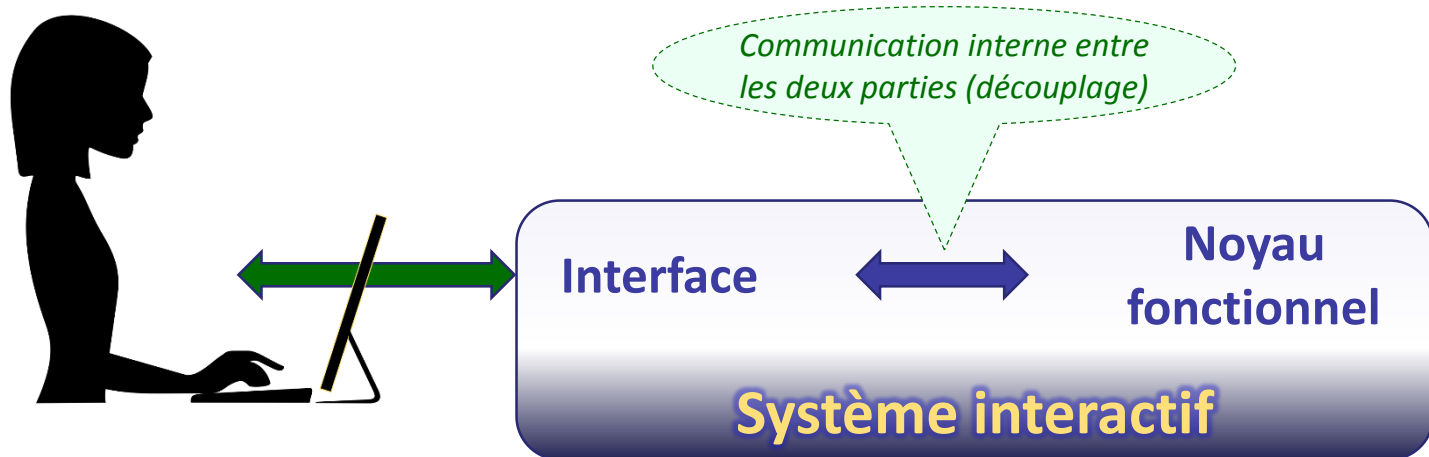


- Les principes de base qui seront abordés s'appliquent à des systèmes destinés à être utilisés dans de nombreux domaines applicatifs, par exemple :
  - Domaine commercial
  - Domaine technique
  - Jeux
  - Réseaux sociaux
  - Didacticiels
  - ...
- Des éléments spécifiques peuvent intervenir en fonction du domaine applicatif et la pondération des contraintes à respecter peut varier, mais les règles de base de l'ergonomie logicielle s'appliquent à tous les systèmes interactifs.

# Systèmes interactifs [3]



- Dans un système interactif on peut considérer deux parties :
  - L'**interface utilisateur** qui englobe tous les éléments matériels et logiciels dédiés à la capture des entrées de l'utilisateur et à la restitution des sorties du système.
  - Le **noyau fonctionnel** qui contient tout le reste et notamment les composants qui effectuent des traitements, des calcul, qui gèrent les informations et le stockage des données, communiquent avec d'autres système, etc.



# Conception de l'interaction



- La **conception** des interfaces homme-machine (*interaction design*) vise à **maximiser** ce que l'on nomme **l'expérience utilisateur** (*User-Experience UX*) qui englobe tous les aspects perçus par l'utilisateur dans l'utilisation du 'produit' (application, site web, etc.).
- Comme toute tâche de conception, cette phase fait appel à une bonne dose de **créativité** mais elle doit être appuyée par de solides connaissances ergonomiques et techniques.
- La conception (*design*) se réfère au processus créatif visant à créer quelque chose de nouveau (l'interface utilisateur) et aboutit à un résultat qui représente l'interface créée sous la forme d'une esquisse, d'un scénario, d'une maquette, d'un prototype (de basse, moyenne ou haute fidélité).
- La conception est rarement un processus linéaire et implique généralement de multiples itérations, des variantes, des phases de validation, etc.



# L'expérience utilisateur UX



- **Une définition de l'expérience utilisateur** [Nielsen/Norman] avec quelques éléments importants mis en évidence :
  - *"User experience" encompasses all aspects of the end-user's interaction with the company, its services, and its products.*

*The first requirement for an exemplary user experience is to meet the exact needs of the customer, without fuss or bother. Next comes simplicity and elegance that produce products that are a joy to own, a joy to use. True user experience goes far beyond giving customers what they say they want, or providing checklist features. In order to achieve high-quality user experience in a company's offerings there must be a seamless merging of the services of multiple disciplines, including engineering, marketing, graphical and industrial design, and interface design.*

# Métaphores [1]



- L'ordinateur ne fonctionne de manière assez différente des humains et ses données, ses actions ne sont pas directement visibles.
- Il est donc important que l'utilisateur puisse se faire une image, une **représentation mentale** (un modèle) du fonctionnement du système et du logiciel qu'il utilise.
- Un bon moyen d'offrir une interface qui permette à l'utilisateur de se forger une représentation mentale (un modèle du comportement) est de tenter d'**imiter le fonctionnement et les réactions de certains objets du monde réel** (que l'utilisateur est sensé bien connaître).
- On parle alors de **métaphore**, c'est-à-dire qu'on crée des éléments graphiques qui ont une **analogie** (présentation, comportement) **avec des objets du monde réel**.

# Métaphores [2]



- Les systèmes d'exploitation utilisent largement la notion de métaphore.
- La **métaphore du bureau** est largement utilisée dans les interfaces graphiques (gestionnaires de fichiers, utilitaires généraux, ...). On représente et manipule des objets qui imitent le travail au bureau (dossiers, documents, poubelles, boîtes aux lettres, carnets de notes, agendas, horloges, etc.).



# Look-and-Feel [1]



- La notion de **look-and-feel** (**L&F**) que l'on nomme parfois *style*, *skin*, *thème*, ... caractérise l'ensemble des **aspects visuels** de l'interface graphique et de ses composants (graphisme, formes, icônes, couleurs, textures, ombres, polices de caractères, espacements, ...).
- Certains aspects du **comportement** de l'interface font également partie de la notion de *look-and-feel*.
  - Manière de sélectionner un élément
  - Comportement lors de déplacement (*drag&drop*)
  - Déclenchement de menus contextuels
  - Changements d'icône du curseur, apparition de bulles d'aides, etc.
- Par extension, on inclut également les éventuels **éléments sonores** de l'interface (retour de saisie, de sélection, de succès, d'erreurs, etc.).
- On offre parfois à l'utilisateur la possibilité de changer le *L&F* et de l'adapter à ses préférences personnelles.

# Look-and-Feel [2]



- La plupart des systèmes d'exploitation disposent de guides de style (*style guides*) qui donnent des directives concernant le *look-and-feel* des applications qu'ils hébergent.
- Ces guides peuvent être très détaillés et donner des indications très précises sur la manière de développer l'interface des applications (couleurs, espacements, alignements, etc.).

