

IMN638 - Interactions visuelles numériques

Chapitre 3.1 - Interaction Homme-machine

Université de sherbrooke

25 novembre 2013

Sommaire

- Paradigmes d'interaction
- Principes et modèle de l'interaction
- Principes d'utilisabilité
- Interfaces et psychologie de la Gestalt

Introduction

En tant que programmeur, il est souvent facile d'oublier l'objectif premier du logiciel conçu. En effet, la très grande majorité des décisions prises par un programmeur analyste lors du développement d'un logiciel concernent l'architecture et l'organisation du code. Cependant, il ne faut pas perdre de vue qu'un logiciel informatique sert d'abord et avant tout d'outils informatique pour l'accomplissement d'une tâche.

Cette tâche, bien souvent, est exposée à un utilisateur ne disposant pas du temps ou des compétences techniques requises pour avoir un degré de compréhension du programme équivalent à celui du développeur. Afin de rendre l'application accessible, il est donc important de produire une interface permettant l'interaction entre l'utilisateur et le logiciel. Cette interface **doit être développée en ayant comme centre de préoccupation l'utilisateur et son expérience avec l'application.**

Sans être exhaustif, le chapitre qui suit enseignera les concepts de base de l'interaction Homme-machine et de la conception d'interface.

Paradigmes d'interaction

Paradigmes d'interaction

Les paradigmes d'interaction sont les principales familles d'interaction homme-machine étant observées dans le monde informatique. Formellement, un paradigme d'interaction est un **ensemble cohérent de techniques d'interaction qui coopèrent de façon étroite ou reposant sur les mêmes principes techniques ou conceptuels.**

Les paradigmes d'interaction permettent de classifier les modes d'interaction en catégories afin d'organiser et caractériser correctement l'étude propre à un mode d'interaction en particulier.

Paradigmes d'interaction

On note quatre paradigmes d'interaction principaux. De la combinaison des différents paradigmes émergent des modes d'interaction plus complexes et des dispositifs d'interaction particuliers.

- Informatique à grande échelle
- Informatique personnelle
- Informatique en réseau
- Informatique mobile

Paradigmes d'interaction

Informatique à grande échelle :

Les premiers ordinateurs ont donné lieu au paradigme d'interaction dit d'**informatique à grande échelle**. Ce type d'interaction se caractérise par le fait qu'il est **impersonnel**. Autrement dit, l'utilisateur n'interagit pas ou très peu avec l'ordinateur.

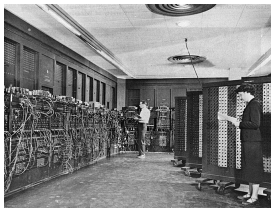
L'informatique à grande échelle est le premier paradigme d'interaction ayant vu le jour. Il est naturellement issu des premiers ordinateurs, où un même ordinateur était souvent partagé entre tous les ingénieurs d'une entreprise ou tous les chercheurs d'une université. Les tâches à effectuer étaient soumises par l'utilisateur à un technicien, qui s'occupait d'exécuter la tâche pour ensuite livrer les résultats à l'utilisateur.

Dans ce cas précis, l'utilisateur n'est pas en contact direct avec l'ordinateur, on dit que l'interaction Homme-machine est impersonnelle.

Informatique à grande échelle (suite) :

L'informatique à grande échelle existe encore aujourd'hui, principalement dans le cas de super-ordinateurs. (Serveurs d'entreprise, ordinateurs s'occupant du réseau internet, super-calculateurs, etc.)

Dans tous les cas, l'informatique à grande échelle demeure **impersonnelle** et **l'interaction est partagée** par tous les utilisateurs du dispositif informatique.



Ordinateur ENIAC (Electronic Numerical Integrator Analyser and Computer) dévoilé en 1946, il est le premier ordinateur entièrement électronique construit pour être Turing-complet. Il est un exemple de dispositif d'interaction informatique à grande échelle. (Université de Pennsylvanie)



Superordinateur Mammouth de l'université de Sherbrooke. Un exemple moderne de dispositif d'interaction informatique à grande échelle. (Université de Sherbrooke)

Paradigmes d'interaction

Informatique personnelle :

Le paradigme d'interaction de l'informatique personnelle est le **résultat d'un décalage de paradigme** par rapport à l'informatique à grande échelle.

En effet, l'informatique à grande échelle a graduellement donné place à des ordinateurs moins couteux, accessibles à un public ayant parfois peu ou aucune connaissances techniques.

La vision de scientifiques tels que **Douglas Engelbart** a permis l'invention de dispositifs d'interactions comme la souris et l'écran d'ordinateur. Ces dispositifs ont à leur tour donné lieu à l'apparition d'ordinateurs dits « personnels ». L'interaction avec ces ordinateurs diffère de l'informatique à grande échelle dans la mesure où l'utilisateur interagit directement, et personnellement avec celui-ci.

Informatique personnelle (suite) :

L'informatique personnelle a donné lieu à l'apparition **deux architectures d'interactions, soit :**

Les ordinateurs de bureaux : Architecture d'interaction ayant été définie par les environnement de travail dans les bureaux. Consiste en un dispositif de pointage (souris, trackball, etc.), un dispositif d'affichage, des logiciels et des dispositifs de stockage (CD, disquette, clé usb, etc.).

Les ordinateurs personnels publics : Architecture d'interaction prenant la forme d'ordinateurs à accès public (ordinateurs de bureau avec mot de passe par exemple) ou d'ordinateurs de service publics (guichets automatiques, téléphones publics). Les dispositifs d'interaction pour cette architecture d'interaction sont souvent physiquement intégrés dans l'ordinateur afin d'éviter qu'ils ne soient endommagés ou volés. (Écran tactiles, claviers fixés à l'ordinateur, etc.)

Paradigmes d'interaction

Informatique en réseau :

L'informatique personnelle est bien adaptée pour un environnement simple, par exemple dans une petite entreprise ou une maison. Par contre, elle reste limitée dans la mesure où elle n'intègre pas la notion de connectivité entre les ordinateurs. Pour étendre les possibilités d'interaction de l'informatique personnelle, l'informatique en réseau a vu le jour.

L'idée de construire un réseau d'ordinateurs permettant l'interaction entre plusieurs utilisateurs provient d'une série de mémots écrits par J. C. R. Licklider en 1962, élaborant le concept de "réseau galactique".

Contrairement aux autres paradigmes d'interaction, l'informatique en réseau n'affecte pas outre mesure la méthode d'interaction entre l'Homme et la machine, la plupart de cette interaction s'effectuant à travers les outils interactifs déjà développés pour l'informatique à grande échelle ou l'informatique personnelle.

Informatique en réseau (suite) :

Ceci étant dit, l'informatique en réseau affecte la notion de moment et d'endroit pour l'interaction. Par exemple, il est possible d'accéder à n'importe quelle ressource d'une entreprise, partout à travers le monde, tant qu'une connexion internet est disponible.

Le paradigme d'informatique en réseau est d'autant plus particulier que l'évolution des besoins interactifs des utilisateurs tend à affecter la manière dont les réseaux se développent. (On pense par exemple aux différents standards de communication cellulaire, 2G, 3G, etc. développés en fonction du type d'interaction requis par les utilisateurs de ces réseaux.)



Le Xerox Alto, introduit en 1973, fut l'un des premiers ordinateurs de bureau à supporter l'utilisation de réseaux informatiques. (Centre de recherche Palo Alto).

Paradigmes d'interaction

Informatique mobile :

Les données deviennent de l'information lorsqu'elles sont organisées de manière sensée et utile. Les connaissance sont quant à elles issues de l'information.

L'accès informatique à des données est limité par l'endroit où se trouvent les ressources informatiques et l'utilisateur. Autrement dit, on observe une lacune dans les paradigmes précédents dans la mesure où un utilisateur n'étant pas à proximité d'un ordinateur n'est pas en mesure d'accéder à ses données, que l'ordinateur soit en réseau ou non, personnel ou non.

Cette lacune tend à être comblée via le paradigme d'interaction de l'informatique mobile. L'informatique mobile englobe les dispositifs d'interaction informatiques suffisamment compacts pour être transportés par l'utilisateur. L'utilisateur étant toujours en présence de son ordinateur mobile, il peut théoriquement toujours accéder à des données.

Informatique mobile (suite) :

D'un point de vu d'interaction pure, l'informatique mobile s'apparente difficilement à l'informatique personnelle dans la mesure où les dispositifs interactifs propres à un ordinateur de bureau ne sont habituellement pas adaptés pour l'informatique mobile (la présupposition d'une surface de travail n'étant plus valide pour l'informatique mobile). On observe donc le développement de dispositifs d'interactions adaptés pour l'informatique mobile. (Pavés tactiles, manches isométriques, roulettes).



Les ordinateurs portables de type "Tablet PC" se présentent sous la forme d'un écran tactile et d'un stylet sont un exemple de dispositif d'interaction informatique mobile. (Philips)



Pour le paradigme d'informatique mobile, les dispositifs interactifs doivent être adaptés. Par exemple, le dispositif de pointage doit fonctionner sans surface, la souris est donc souvent remplacée par un pavé tactile. (Wikipedia)

Paradigmes d'interaction

Les différents paradigmes d'interaction, lorsque combinés, donnent lieu à l'apparition de **modèles d'interaction** particuliers. Parmi les plus importants, on note :

- **Informatique collaborative** : Permet à plusieurs utilisateurs d'interagir pour travailler en collaboration (sur les mêmes ressources, au même moment). (Personnel, Réseau, Mobile).
- **Informatique ubiquitaire** : L'interaction avec l'ordinateur se fait de manière transparente à travers l'environnement naturel de l'utilisateur. (Personnel, Réseau, Mobile).
- **Réalité virtuelle** : Inverse de l'informatique ubiquitaire. L'interaction avec l'ordinateur s'effectue en remplaçant l'environnement naturel de l'utilisateur par un environnement virtuel. (Réseau, Mobile).

Paradigmes d'interaction :

Les paradigmes d'interaction et les différents modèles d'interaction auxquels ils donnent lieu peuvent être représentés sous la forme d'un diagramme.

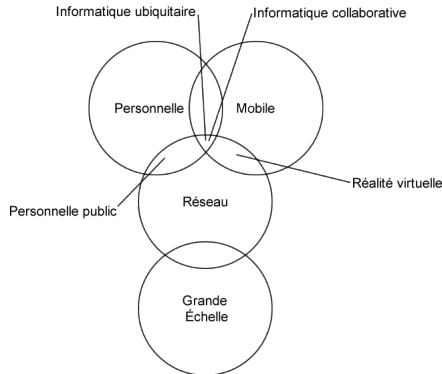


Diagramme représentant les différents paradigmes (cercles) et les modèles d'interaction s'y rattachant. On remarque que les paradigmes se combinent afin de donner différents modèles d'interaction.

Modèle de l'interaction

Modèle de l'interaction

Pour mieux être défini, le concept d'interaction entre l'Homme et un dispositif quelconque (objet, ordinateur, etc.) peut être modélisé à l'aide du **cycle d'exécution/évaluation d'action**.

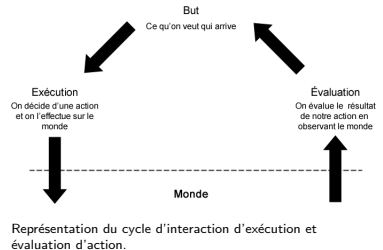
Ce modèle de l'interaction a été proposé par Donald Norman en 1990, dans le livre *The design of Everyday Things*. Il se divise en 4 parties différentes :

- **But** : Nous avons une certaine idée de ce que nous voulons qui arrive avec le dispositif. Cette idée est notre but.
- **Exécution** : Nous décidons quelle action effectuer sur le monde et on l'effectue.
- **Monde** : Nous exécutons les actions sur le monde. (Par exemple manipuler un objet, appuyer sur un bouton, etc.)
- **Évaluation** : Nous validons l'action effectuée en comparant le résultat de l'action sur le monde par rapport à notre but.

Modèle de l'interaction

L'accomplissement d'une tâche par un utilisateur du programme se traduit habituellement par plusieurs itérations du cycle d'interaction. Un système interactif bien construit sera un système où chaque partie du cycle d'interaction sera définie de façon claire et compréhensible pour l'utilisateur à tout moment de son interaction avec le système.

Notons que le cycle d'interaction n'a pas nécessairement de début ou de fin pour une séquence d'interaction. Par exemple, une action effectuée en réaction à un évènement débutera au niveau de l'évaluation, alors qu'un objectif à long terme débutera souvent de manière planifiée au niveau du but.



Gouffre d'exécution et d'évaluation

Lors d'une session d'utilisation d'un système interactif, l'utilisateur peut avoir de la difficulté à accomplir le cycle d'interaction. En supposant que l'utilisateur ne possède pas de problème cognitif, cette difficulté est souvent due à un **gouffre d'exécution** ou un **gouffre d'évaluation** causé par une faiblesse de notre système interactif.

Gouffre d'exécution et d'évaluation

Gouffre d'exécution :

Un gouffre d'exécution survient lorsque le modèle mental que s'est fait l'utilisateur quant aux actions à accomplir pour effectuer une tâche diffère des actions réelles à accomplir.

Lorsque l'utilisateur est en difficulté, dans un gouffre d'exécution, il effectuera habituellement des actions inutiles, des actions ne lui permettant pas d'atteindre l'objectif ou l'utilisateur n'effectuera simplement pas d'action. Ceci est dû au fait que le système qui lui est présenté ne se retrouve pas dans les états attendus par l'utilisateur après que celui-ci ait effectué une action.

Pour remédier à un gouffre d'exécution, on doit modifier le système de manière à ce que les séquences d'action requises pour atteindre un objectif soient bien celles attendues par l'utilisateur.

Exemple de gouffre d'exécution :

Un utilisateur souhaite sauvegarder un document. Pour sauvegarder le document, il sait qu'il doit aller dans le menu "fichier" puis ensuite sélectionner "enregistrer sous".

Si le menu ouvre et qu'il n'y a pas d'option "enregistrer sous" dans celui-ci, l'utilisateur devra cesser le cycle d'interaction et réévaluer ses actions, pour potentiellement rebrousser chemin.

On a là un cas typique de gouffre d'exécution. L'utilisateur pensait qu'en appuyant sur "fichier" il apparaîtrait l'option "sauvegarder" alors que ce n'est pas le cas. Il y a donc un gouffre d'exécution entre ce qui est attendu par l'utilisateur et ce qui arrive réellement dans le système interactif.

Gouffre d'exécution et d'évaluation

Gouffre d'évaluation :

Un gouffre d'évaluation est un problème similaire au gouffre d'exécution, mais survenant lors de l'évaluation de notre système par l'utilisateur.

Pendant l'évaluation, l'utilisateur évalue l'état du système courant afin de fixer son prochain objectif (ou sous-objectif) et ainsi continuer son interaction. Si l'utilisateur n'est pas en mesure de localiser son prochain objectif dans l'interface, ou qu'il n'arrive pas à évaluer de manière claire quel a été l'effet de son action sur le système, il ne sera pas en mesure de continuer son interaction.

Plus l'occurrence de gouffres d'évaluation est importante, ou plus les gouffres d'évaluation sont graves, plus l'effort cognitif requis pour utiliser le système interactif sera grand et moins l'expérience d'utilisation sera positive.

Gouffre d'évaluation (suite) :

Dans certains cas, le gouffre d'évaluation sera intrinsèque au dispositif utilisé. Par exemple, si on utilise un menu déroulant dans une application, il existe un gouffre d'évaluation causé par le fait que le contenu du menu déroulant n'est pas visible en tout temps. L'utilisateur devra alors choisir le menu déroulant qu'il croit le meilleur afin d'atteindre la fonctionnalité dont il a besoin, courant le risque de devoir essayer plusieurs menus différents avant d'atteindre le résultat voulu.

Dans de tels cas, les gouffres d'évaluation sont nécessaires puisque l'exposition de la totalité des fonctionnalités du système d'un seul coup deviendrait beaucoup trop lourde pour l'utilisateur.

Gouffre d'évaluation (suite) :

Un autre exemple de gouffre d'évaluation est l'utilisation de "modes" qui ne sont pas clairement définis dans un système.

Par exemple, un clavier en mode "verrouillage majuscule" (caps lock) n'indique pas toujours clairement qu'il se mettera à taper uniquement en majuscules (rien à cet effet n'est affiché à l'écran, seulement une petite lumière sur le clavier s'allume).

Ceci peut confondre un utilisateur novice n'ayant pas réalisé qu'il avait mis le clavier dans ce mode.

Ceci crée donc un gouffre d'évaluation, l'utilisateur n'arrivant pas à évaluer que le clavier est en mode "écrire seulement en majuscules", devenant alors confus quant au fait que le clavier n'agit pas comme à l'habitude.

Gouffre d'évaluation (suite) :

Pour réduire au maximum les probabilités de gouffre d'évaluation, Donald Norman propose certaines questions pour "vérifier" si notre système évite ces derniers :

Selon un design particulier, à quel point pouvez vous facilement ...

- déterminer la fonction d'un dispositif ?
- déterminer quelles actions sont possibles ?
- déterminer comment se traduit une action sur le système ?
- effectuer l'action ?
- déterminer si le système est dans l'état désiré ?
- déterminer comment fonctionne l'état courant du système ?
- déterminer dans quel état est le système ?

Si, pour une de ces questions, la réponse est "difficilement", le système doit être modifié afin qu'il soit possible de répondre favorablement à la question.

Principes d'utilisabilité

Principes d'utilisabilité

Le design de systèmes interactifs, comme tout autre domaine de design, est guidé par des principes bien établis. Ces principes sont des lignes directrices pour la conception d'un système interactif correct.

Ces principes, nommés les "principes d'utilisabilité" furent proposés Jakob Nielsen en 1994 et touchent les différentes facettes d'un système interactif. On les compte au nombre de six :

- Prédictibilité (ou fiabilité)
- Facilité d'apprentissage
- Efficacité
- Facilité de mémorisation (ou d'appropriation)
- Satisfaction subjective
- Facilité de compréhension

Principe de prédictibilité (ou fiabilité)

Le principe de prédictibilité, parfois appelé "principe de fiabilité", et le principe selon lequel les résultats des actions effectuées dans un système sont prédictibles.

La notion de prédictibilité s'effectue sur deux plans :

La prédictibilité dans le temps : Chaque fois qu'une action est effectuée, elle a le même résultat sur le système. Le résultat ne change pas aléatoirement de fois en fois.

La prédictibilité à priori : Même si l'utilisateur n'a jamais effectué une action, le vocabulaire employé et l'organisation des éléments du système permettent de prévoir correctement le résultat des actions. *(Par exemple, lorsque l'utilisateur clique sur un icône de disquette, il s'attend à ce que le document soit sauvegardé, et non pas à voir son lecteur éjecter la disquette.)*

Principe de facilité d'apprentissage

La facilité d'apprentissage, comme son nom l'indique, est le principe selon lequel l'utilisateur apprendra facilement à se servir des fonctionnalités particulières à un système interactif spécifique. La facilité d'apprentissage dépend directement de la facilité de compréhension. Plus un système interactif est facile à comprendre, plus il sera facile à apprendre.

L'utilisation des connaissances à priori de l'utilisateur (connaissances avant d'utiliser l'application pour la première fois) est le meilleur moyen de faciliter l'apprentissage d'une application. Ces connaissances à priori sont souvent issues de son expérience préalable avec des systèmes similaires ou des objets de la vie courante. Utiliser les mêmes standards, le même vocabulaire ou le même fonctionnement que ces derniers une méthode simple pour faciliter l'apprentissage de votre propre système.

(Par exemple, utiliser un symbole de loupe pour agrandir une partie de l'interface utilise un objet courant que connaît l'utilisateur, permettant à ce dernier d'apprendre rapidement la fonctionnalité du système.)

Principe d'efficacité

Le principe d'efficacité englobe tout ce qui touche à la rapidité d'accomplissement des tâches au sein d'un système interactif. Globalement, l'effort et le temps requis pour l'accomplissement d'une tâche dans un système doit être réduite autant que possible, afin que l'utilisateur puisse utiliser celui-ci de façon efficace. Le principe d'efficacité se traduit par différents sous-concepts :

- Flexibilité du système : Si l'utilisateur peut adapter le système à ses propres exigences, le système devient plus efficace.
- Performances du système : Les actions entreprises par le système sont effectuées rapidement.
- Stabilité du système : Le système fonctionne toujours correctement, l'utilisateur ne perd pas de temps à récupérer des erreurs du système.
- Simplicité du système : Une action simple ou une action courante s'effectue plus rapidement.

Principe de facilité de mémorisation

Le principe de facilité de mémorisation (ou d'appropriation) décrit la facilité avec laquelle un utilisateur pourra mémoriser les séquences d'actions à effectuer pour accomplir un objectif. Contrairement au principe de facilité d'apprentissage, on ne vérifie pas ici à quel vitesse l'utilisateur apprend à effectuer des actions, mais plutôt avec quelle facilité il mémorise des séquences d'actions une fois qu'elles ont été apprises.

La mémorisation peut être favorisée en construisant un système interactif intuitif pour l'utilisateur. Notamment en utilisant des images ou des icônes ou en plaçant spatialement les composantes du système de façon logique. Similairement au principe de facilité d'apprentissage, l'utilisation de séquences d'actions déjà mémorisées dans la vie courante ou dans un système similaire peut aussi faciliter la facilité de mémorisation.

(Par exemple, il est facile de se rappeler que pour supprimer un document, l'utilisateur doit le glisser dans la corbeille, puisque c'est la séquence d'action qu'il effectue normalement dans la vie courante.)

Principe de satisfaction subjective

Le principe de satisfaction subjective est un principe difficile à caractériser et à maîtriser dans la mesure où la satisfaction subjective peut changer d'un utilisateur à l'autre.

En moyenne, certains concepts d'interaction demeurent relativement universels et permettent une satisfaction pour la plupart des utilisateurs.

La simplicité d'une interface et sa qualité esthétique sont des exemples de facteurs pouvant augmenter la satisfaction subjective d'un utilisateur.

Similairement, les utilisateurs d'un système se sentent habituellement satisfait de leurs actions lorsque ces dernières ont un impact observable sur le système.

(Par exemple, lorsque l'utilisateur décide de vider sa corbeille, la corbeille se vide et apparaît vide plutôt que chargée de papiers, ceci satisfait mieux utilisateur que s'il n'y avait aucune différence entre la corbeille vide ou pleine.)

Principe de facilité de compréhension

Le dernier principe, le principe de facilité de compréhension, englobe la facilité à comprendre comment fonctionne un système interactif. Cette compréhension se traduit par la bonne compréhension des différentes fonctionnalités du système, de leur effet sur le système, des contraintes de celui-ci, etc.

La méthode principale pour faciliter la compréhension d'un système est d'utiliser des métaphores. Une métaphore d'un point de vu interactif consiste à reproduire virtuellement le fonctionnement d'un objet réel de manière à faciliter la compréhension d'un concept virtuel similaire.

Un exemple classique de métaphore est la représentation du concept de document informatique tel que représenté dans le logiciel Microsoft Word. Le document est visuellement représenté comme une série de pages, ce qui permet à l'utilisateur de comprendre plus facilement l'effet de ses actions sur le document, puisque ce qu'il voit correspond à ce qu'il obtiendra lorsque le document informatique sera imprimé.

Interfaces et psychologie de la Gestalt

Interfaces et psychologie de la Gestalt

L'interaction Homme-machine, dans un contexte informatique, s'effectue habituellement à travers des interfaces graphiques. Ces derniers sont donc plus souvent qu'autrement au centre de la recherche effectuée en interaction.

De cette recherche, certains principes de base gouvernant l'interaction en générale ont été émis. Parmi ces principes, nous avons vu les principes d'utilisabilité, qui dictent les lignes directrices de la conception de l'interaction d'un point de vu fonctionnel.

Similairement, certains principes sont suggérés pour la construction d'interfaces visuels. Ces principes tendent à favoriser une meilleure interaction avec les interfaces visuelles en encourageant une bonne organisation spatiale de l'interface. Ces principes sont présentés et étudiés dans ce qu'on appelle **la psychologie de la forme, ou psychologie de la Gestalt**.

Interfaces et psychologie de la Gestalt

La psychologie de la Gestalt est une théorie permettant de dresser un lien entre la présentation d'une forme de manière formelle et la représentation dont se fait le cerveau de cette forme. Autrement dit, la psychologie de la Gestalt explique la façon dont le cerveau humain classifie et regroupe des éléments distincts.

Ceci permet alors de concevoir des interfaces plus efficaces dans la mesure où ces interfaces utilisent à leur avantage la tendance du cerveau à organiser les formes qu'il observe.

On note trois principes et six lois principales issues de la psychologie de la Gestalt. Une interface appliquant ces principes et ces lois est habituellement plus simple à utiliser pour le cerveau humain, et offre par le fait même une meilleure expérience d'interaction.

Principes de la Gestalt

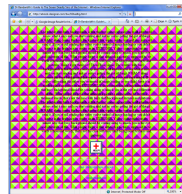
Les principes de la Gestalt sont :

- Il n'existe pas de perception isolée, la perception est initialement structurée.
- La perception consiste en une distinction de la figure sur le fond.
- Le tout est perçu avant les parties le formant.

La structuration des formes ne se fait pas au hasard, mais selon certaines lois dites "naturelles" et qui s'imposent au sujet lorsqu'il perçoit.



Vase de Rubin, la perception du fond et de la figure peut changer, ce qui affecte notre perception de l'image. (Vase ou 2 personnes qui se font face.)



Il est difficile de lire le texte d'un site web lorsque le fond et la forme ont un niveau de détail comparable. La perception du texte est donc très difficile.



Le smiley est plus facilement reconnaissable lorsque assemblé pour former un tout, que lorsque seulement ses parties sont vues.

Lois de la Gestalt

De ces principes, on extrait six lois principales, soit :

- Loi de clôture
- Loi de similitude
- Loi de proximité
- Loi de symétrie
- Loi de continuité
- Loi de mouvement parallèle

Nous verrons chacune des lois en détail dans les acétates qui suivent.

Loi de clôture

La loi de clôture décrit la tendance du cerveau à former des éléments finis à partir de stimuli non perçus ou incomplets. Elle fait ressortir la tendance du cerveau à augmenter la régularité des formes afin de simplifier leur représentation.

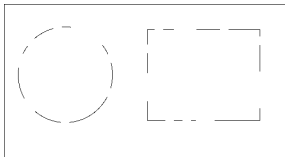


Illustration de la loi de fermeture. Notre cerveau voit ici un cercle et un rectangle côte à côte, alors que ces formes ne sont pas explicitement définies.



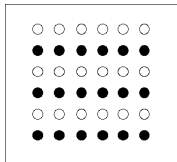
Le regroupement en bas à droite a tendance à être perçu comme un groupe compact par rapport aux autres éléments de l'image, car il est composé de manière à former un rectangle, ce qui illustre encore le principe de fermeture.



Loi de fermeture utilisée dans un contexte réel. La bannière du haut de la page n'est en réalité qu'une série d'images séparées. Sa structure donne cependant l'impression d'une bannière fermée et complète. (Usherbrooke.ca, 2008)

Loi de similitude

La loi de similitude est posée telle que des éléments possédant les mêmes caractéristiques visuelles, comme la forme, la couleur ou la grandeur, seront associés, sans que leur positionnement spatial n'entre en ligne de compte.



Notre cerveau a tendance à regrouper les cercles noirs et les cercles blancs en différentes rangées, malgré le fait que les cercles soient uniformément répartis dans l'image. On a là un exemple de la loi de similitude.



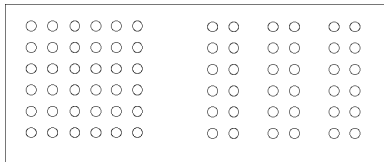
Même sans organisation spatiale, notre cerveau a tendance à regrouper les rectangles de couleur et de forme similaires, un autre exemple de la loi de similitude.



La loi de similitude peut être utilisée concrètement afin de regrouper des éléments au sein d'une interface. Dans ce cas-ci, trois images similaires (carré orange dans un carré blanc) sont perçues comme étant un même groupe de liens même si elles sont relativement distancées les unes des autres et possèdent des dimensions variables. (Ethias.be, 2009)

Loi de proximité

La loi de proximité est la loi selon laquelle le cerveau tend à regrouper les éléments spatialement ou temporellement à proximité. Dans un contexte d'interface, la proximité des éléments crée des regroupements logiques de ces derniers.



Les cercles, lorsque regroupés de manière à avoir une proximité variable, font apparaître des groupes bien démarqués, malgré l'uniformité de leur forme prise individuellement. (On remarque les groupes même si tous les cercles sont pareils.)



Regrouper les éléments sémantiquement ou fonctionnellement similaires en utilisant le principe de proximité permet de naviguer à travers une interface plus rapidement. Un exemple concret et classique de l'utilisation de la loi de proximité est le regroupement de boutons. Dans ce cas-ci, les boutons permettant le contrôle de la fenêtre sous Windows. (Microsoft, 2009).

Loi de symétrie

La loi de symétrie décrit la tendance du cerveau à percevoir collectivement un regroupement d'éléments symétriques, même si ces éléments sont distancés les uns des autres.



Représentation schématique du principe de symétrie. Les éléments en couleur sont des composantes d'interface tirant profit du principe de symétrie. Malgré le fait qu'elles soit intégrées avec une multitudes d'autres composante, leur symétrie relative permet de les regrouper de manière logique.



Exemple concret d'utilisation du principe de symétrie. Les boîtes "sports" (rouge) et "divertissement" (doré) ne semblent pas reliées à priori. Elles sont à distance égale des autres composantes d'interface et n'ont pas la même couleur. Cependant, leur structure et leur disposition est symétrique, ce qui explique pourquoi nous avons tendance à les associer dans la même catégorie de contenu. (Radio-canada.ca)

Loi de continuité

La loi de continuité met en valeur le fait que le cerveau possède une tendance à compléter les signaux visuels, auditifs ou cinétiques en des trajectoires continues, même si le signal n'est pas lui-même continu. D'un point de vue visuel, ceci se traduit par la tendance du cerveau à créer des trajectoires continues où il n'y en a pas. Les éléments sur cette trajectoire sont perçus comme étant dans un même regroupement logique.



Le cerveau crée des trajectoires à partir des cercles même si les trajectoires ne sont pas explicitement définies. De plus, nous avons tendance à considérer la trajectoire en arc de cercle composée de trois cercles gris et les cinq cercles noirs avant la trajectoire formée des cercles gris, car la première est plus simple. C'est là une représentation de la loi de continuité.



Une trajectoire est ici formée entre les éléments non connexes car ils sont alignés sur une même ligne. C'est un autre exemple de continuité.



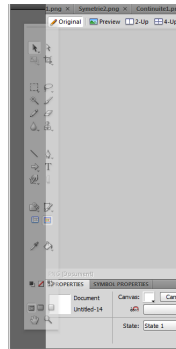
Cas pratique où la notion de continuité est appliquée. Les trois boutons du site internet sont associés ensemble car ils forment une même trajectoire, même s'ils sont de taille différentes et qu'ils abordent des images différentes.

Loi de mouvement parallèle

La loi de mouvement parallèle est une loi s'appliquant aux objets mobiles. Globalement, elle décrit la tendance qu'à le cerveau humain à associer ensemble les objets se déplaçant en groupe, à une vitesse similaire.



Les objets se déplaçant ensemble, à la même vitesse, sont perçus comme un groupe avant d'être perçus individuellement. C'est le cas par exemple pour une formation d'avions volant dans le ciel. (Gettys Images)



Mouvement parallèle appliqué dans un contexte d'interface informatique. Un panneau d'outils est détaché de la barre d'outils. Tous les outils qu'il contient bougent ensemble, ils sont donc associés au même groupe par le fait même.

Conclusion

Conclusion

Notons qu'il n'y a pas de formule ultime pour construire une interface ou un système interactif parfait. La construction d'un bon système interactif se fait en tenant compte de la psychologie et des capacités de l'utilisateur visé par le système.

Ce chapitre n'est qu'une brève introduction à l'interaction Homme-Machine. Il offre cependant des notions élémentaires à mettre en pratique lors de la conception d'un système interactif. Savoir appliquer les différents principes et les différents modèles vus dans ce chapitre dans un contexte réel constitue un très bon point de départ pour concevoir correctement un logiciel interactif.

Pour terminer, notons qu'il sera presque toujours impossible d'éliminer totalement les faiblesses d'interface ou d'interaction dans une application informatique. Le défi principal lors de la conception d'un système interactif est de réduire au maximum l'impact de ces faiblesses tout en garantissant une expérience plaisante pour l'utilisateur.

Références suggérées

- Livre** Steven Heim, "*The Resonant Interface, HCI Foundations for interaction design*", Addison Wesley, 2007, pp. 12 à 232
- Livre** Donald A. Norman, "*The Design of Everyday Things*", Basic Books, 2002