

# Table des matières

- Table des matières
- I. Présentation du sujet et délimitation du périmètre
- I. a. Introduction
- I. b. Définition d'une interface homme-machine
- II. a. La technique et son évolution historique
- II. a.1 Définition des moyens d'interaction historique (Antiquité - 1960)
- II. a.2 Terminal et interactivité
- II. a.3 Apparition et développement des graphismes
- Terminaux graphiques vectoriels
- Terminaux graphiques matriciels
- Sketchpad et les interfaces graphiques
- II. a.5 Xerox et spécification des interfaces modernes
- Xerox ALTO
- Xerox Star
- Les standards d'interaction
- II. a.6 Bibliothèques et organisation technique de l'affichage
- Le système de fenêtrage X
- Les bibliothèques graphiques
- II.b Prendre de la hauteur
- II.b 1 - Considérations et conséquences relatives à ces évolutions
- II.b 2 - Penser son interface
- II.b 3 Accessibilité et design universel
- II.b 4 Efficacité et productivité
- II.b 5 Efficience et cognitive

# I. Présentation du sujet et délimitation du périmètre

## I. a. Introduction

Fondamentalement, il est possible de considérer un ordinateur comme étant une machine capable de transformer des informations données (entrées) en informations traitées (sorties) et de stocker celles-ci à plus ou moins long terme .

En se basant sur une telle description, il est simple de se rendre compte d'une problématique majeure liée au bon fonctionnement de ces appareils:

*Comment transmettre de manière efficace des informations vers et depuis l'ordinateur ? Entre le manipulateur et la machine ?*

Si ce questionnement peut paraître naturel, il faut savoir qu'il se trouve être aujourd'hui au coeur d'une meta-discipline à la croisée des domaines de recherche, entre l'informatique, la neurologie, la cognitive, l'ergonomie et toute une nébuleuse de domaines dont le lien avec l'informatique peut sembler plus ou moins évident: l'étude des interfaces Homme-Machine (IHM ou HCI en anglais).

Il est important de souligner qu'avec le développement des nouvelles technologies, la multiplication de leur formes physiques, caractéristiques et de leur utilisateurs, il devient de plus en plus important, tant d'un simple point de vue pratique qu'économique ou scientifique, d'être en mesure de fournir des environnements d'interaction de qualité à l'ensemble des utilisateurs quels qu'ils soient.

Ainsi, étant donné l'intérêt substantiel de cette problématique, nous allons tâcher, tout au long de ce document, de présenter les diverses évolutions techniques ayant permis aux IHM d'évoluer et de se démocratiser, les différents éléments non-techniques permettant au domaine de continuer d'évoluer pour enfin aborder les problématiques auxquelles les IHM vont se confronter à l'avenir ainsi que leurs potentielles évolutions.

## **I. b. Définition d'une interface homme-machine**

Avant de poursuivre, il est nécessaire de définir précisément ce qu'est une interface homme-machine.

D'un point de vue informatique, une interface est considérée comme suit:

Interface : un dispositif qui permet des échanges et interactions entre différents acteurs - Source : Wikipédia  
(<https://fr.wikipedia.org/wiki/Interface>)

Dans notre cas, nous considérerons comme interface Homme-Machine tout système permettant la transmission d'information l'utilisateur vers la machine et inversement.

Dans les faits et de manière générale, une interface prend la forme d'un terminal disposant de plusieurs dispositifs d'entrées et/ou de sorties. On se situera dans ce document du point de vue de la machine. Une entrée (input) sera donc un ensemble d'informations transmises de l'utilisateur vers la machine tandis qu'une sortie (output) sera un ensemble de données transmises de la machine vers l'utilisateur.

schéma IHM

schéma IHM

## II. a. La technique et son évolution historique

### II. a.1 Définition des moyens d'interaction historique (Antiquité - 1960)

Avec l'apparition et le développement des mathématiques, en tant qu'élément pratique puis en tant que science à part entière, les hommes ont rapidement cherché à faciliter les opérations de calcul pouvant rapidement devenir fastidieuses à mesure que les besoins en précisions et en quantité de données à traiter augmentaient. Concrètement, cette recherche de la rapidité et de la facilité de calcul permit l'invention d'un certain nombre de machines à calcul mécanisées (à partir de 1592) ou non telles que les tables à calcul, les bouliers ou, plus tard les machines de Schickard ou de Pascal (Abaques).

#### Un boulier

#### Un boulier

Si l'interaction avec des machines à calcul non-mécanique est relativement naturelle et simple (puisque c'est le manipulateur qui va directement bouger les éléments du dispositifs permettant le calcul), celle-ci se complexifie dans le cas d'une machine mécanique. En effet, l'action d'un utilisateur doit être transformée en force mécanique permettant au mécanisme de produire un résultat. On note donc l'apparition de roues dentées, de manivelles, puis de boutons voir d'un clavier complet (D.D. Parmelee, 1850) jouant ici le rôle **d'interfaces** de l'utilisateur vers la machine. Et ce, dans le sens où ces dispositifs permettent de transformer une action utilisateur ayant un sens pour ce dernier en éléments "compréhensible" et utilisable par le système de la machine à calculer.

En ce qui concerne la transmissions des informations depuis la machine vers le manipulateur, l'évolution et les problématiques restent relativement similaires. Dans le cas des machines mécanisées, des roues dentées, actionnées par le mécanisme interne et portant des inscriptions présentant les différents éléments des résultats permettent à l'utilisateur de prendre connaissance de la "sortie" de la machine.

Que ce soit concernant les entrées ou les sorties, l'apparition des machines mécanisées marque un tournant. En effet, il devient nécessaire d'être en mesure de transmettre de l'information depuis l'utilisateur vers le système et inversement. C'est donc à partir de celle-ci qu'il est réellement possible de parler d'interfaces homme-machines (IHM), même si celles-ci restent relativement basique.

Une autre révolution dans le domaine des IHM est relative à l'apparition des machines électro-mécaniques et/ou programmables. En effet en 1890, Herman Hollerith développe une machine capable de lire des cartes préalablement perforées. Ici, l'utilisation de l'électricité dans le fonctionnement de la machine permet une abstraction supplémentaire quand à son mécanisme de traitement interne, en rendant la lecture de cartes perforées possible (à noter qu'Hollerith fondera, grâce à son système, la société qui deviendra IBM).

### Machine de Hollerith

#### Machine de Hollerith

Durant la période suivant l'invention des machines électro-mécaniques et jusqu'au début des années 60, on ne distinguera pas spécialement d'évolutions majeures concernant les interfaces homme-machines, celle-ci restant limitées à l'usage de manettes, boutons, lampes, cartes perforées ou listing imprimé que ce soit pour transmettre des sorties ou des entrées (même s'il est possible de souligner l'invention et l'usage de l'imprimante, permettant de conserver les résultats du traitement d'une machine).

A noter que la notion d'interface abordée ci-dessus est plutôt large. En effet, l'échange d'information entre utilisateurs et machines s'est toujours réalisé ici avant et après le traitement. Le processus d'échange permis par ces interfaces n'est donc pas interactif.

## **II. a.2 Terminal et interactivité**

Problématique majeure si l'en est, cette impossibilité d'interaction en temps réel avec la machine fut dépassée avec l'invention et le développement de terminaux.

Basiquement un terminal peut-être défini comme un appareil électronique et électro-mécanique utilisé pour transmettre des données vers une machine, et consulter des données depuis ladite machine<sup>ref\_</sup> ([https://en.wikipedia.org/wiki/Computer\\_terminal](https://en.wikipedia.org/wiki/Computer_terminal)) (Il est aisé de constater qu'on se rapproche ici fortement de la définition même de ce qu'est une interface homme-machine).

Les terminaux font leur apparition entre les années 50 et 60 mais sont dans un premier temps peu utilisés car extrêmement lents comparés à l'usage de cartes perforées ou aux autres technologies existantes en générales.

En effet, les premiers appareils existants consistent en l'alliance d'une imprimante permettant la sortie d'information et d'un clavier, rendant possible l'entrée de données. Ils étaient donc fortement limités en terme de vitesse de transmission de l'information notamment par le fait que les informations étaient transmises à l'utilisateurs à la vitesse d'impression de l'imprimante (qui n'était pas alors des plus performantes).

ASR-33

ASR-33

Une solution envisagée fut alors l'usage d'un écran (type télévision) à tube cathodique pour réaliser une sortie de données nettement plus rapide. Cependant, même si les moyens technologiques des

années 50-60 permettaient en théorie la production de ce type de terminaux, il n'en fut rien. En effet, étant donnée la quantité de données nécessaire au stockage de l'information à afficher sur le terminal et la problématique posée par la synchronisation de l'affichage vidéo et du traitement logique de l'image à afficher, développer ce type de terminal serait revenu beaucoup trop coûteux, tant d'un point de vue financier, que spatial (espace nécessaire à l'intégration de mémoire dans le terminal).

Il faudra attendre le début des années 70 pour que les premières unités d'affichage visuel (VDU, visual display units) fassent leur apparition (on peut citer par exemple le Datapoint 3300 ([https://en.wikipedia.org/wiki/Datapoint\\_3300](https://en.wikipedia.org/wiki/Datapoint_3300))), et avec eux un certain nombre d'innovations telles que les caractères ASCII ou l'invention des ports séries à 25 pin permettant la connexion du terminal à son unité logique (ordinateur).

En France, l'exemple le plus marquant et le plus connu de terminal informatique reste le Minitel, exploité de 1980 à 2012, qui présente la particularité d'être relié à son unité logique (serveur) via réseau téléphonique.

### Minitel

#### Minitel

En terme d'innovation dans le domaine des IHM, l'usage des terminaux fut révolutionnaire. En effet, ils permirent enfin un usage interactif des machines logiques de l'époque et autorisant notamment la machine à proposer à son utilisateur d'entrer des informations pendant l'exécution d'un programme (premiers "champs de texte"), et ce tout en ouvrant la voie au développement de dispositifs graphiques plus évolués.

A noter que l'on distingue deux types de terminaux:



- Les terminaux intelligents, qui disposent de leur propre unité logique.
- Les terminaux légers, qui se connectent à une unité de calcul distante (Le minitel cité plus haut est un parfait exemple de terminal léger).

## **II. a.3 Apparition et développement des graphismes**

L'apparition et le développement des terminaux informatique permet, à terme (1960 puis réel développement à partir de 1970), la création de terminaux graphiques, capable de représenter de nouveaux éléments visuels, en plus de texte.

Ceux-ci reposent initialement sur l'usage d'un écran à tube cathodique, dont la vitesse de rafraichissement permet d'émuler des mouvements ou encore des images persistentes. A noter que c'est cette problématique de vitesse de rafraichissement (et les éléments qu'elle soulève) qui est responsable du développement plus tardif des terminaux graphiques et de leur coût initialement plus élevé.

On distingue ces terminaux graphiques en deux catégories, reposant sur deux philosophies distinctes présentant chacune leurs avantages et inconvénients.

### **Terminaux graphiques vectoriels**

Les terminaux graphiques vectoriels permettent de dessiner, grâce à un jeu d'instructions, des lignes à l'écran. C'est la combinaison de ces lignes de tailles variables qui permet la représentation d'images.

Le principal inconvénient de cette méthode d'affichage est que, étant donné les capacités limitées des appareils de l'époque, seul un nombre limité de lignes pouvait être dessiné à l'écran.

A noter qu'aujourd'hui, sauf cas isolé, ce genre d'affichage est totalement abandonné en ce qui concerne le monde de l'informatique.

Terminal graphique vectoriel

Terminal graphique vectoriel

## **Terminaux graphiques matriciels**

Les terminaux graphiques matriciels reposent eux sur l'idée qu'une image peut-être représentée comme un ensemble de points de couleurs (ou de nuances de couleurs).

Ainsi un écran associé à ce principe matriciel va représenter l'image point par point, line par line. L'image ainsi produite n'est visible en son entièreté qu'un court instant pour l'oeil humain, il est donc nécessaire de rafraichir l'image à haute vitesse afin de donner une image persistante, en se basant sur le principe de persistance rétinienne.

Un autre inconvénient de ce système repose sur le fait qu'une image constituée de cette manière ne peut être agrandie sans perte significative de qualité. Contrairement à un affichage reposant sur l'usage de vecteurs.

# Sketchpad et les interfaces graphiques

L'apparition de moniteurs capables d'afficher des images, associé à l'évolution des capacités des machines en terme de puissance de calcul et de mémoire permet au Pr. Ivan Sutherland, en 1963, de développer le logiciel Sketchpad, fonctionnant sur Lincol TX-2, dans le cadre de sa thèse pour le MIT.

## Sketchpad

### Sketchpad

Les possibilités illustrées par le projet de Sutherland sont gigantesques pour l'époque.

En effet, avec son système Sutherland innove dans le domaine des interfaces homme-machines, en proposant une manipulation graphique changeant radicalement de l'habituel terminal textuel (qui introduit les concepts de manipulation 3D et d'interface graphique) , mais aussi dans le domaine du développement en informatique, étant donné que son système met en oeuvre le paradigme de développement orienté objet alors encore en cours de conceptualisation.

L'apparition de ces nouvelles technologies a ouvert la porte à de nombreuses innovations, toutes axées autour du concept d'interface graphique qui peut être défini dès lors comme un "dispositif de dialogue homme-machine, dans lequel les objets à manipulés sont dessinés [...] à l'écran de sorte que l'utilisateur peut utiliser en imitant la manipulation physique de ces objets"<sup>ref</sup> ([https://fr.wikipedia.org/wiki/Interface\\_graphique](https://fr.wikipedia.org/wiki/Interface_graphique)).

## **II. a.5 Xerox et spécification des interfaces modernes**

Les travaux de Suntherland ayant permis l'ouverture du monde de l'informatique à de nouvelles façons d'interagir avec la machine, les recherches et avancées sur le sujet s'accélérent au début des années 70 avec les travaux du laboratoire PARC de Xerox (pionnier dans le domaine de la production d'imprimantes).

### **Xerox ALTO**

En 1970, la société Xerox crée un laboratoire de recherche le PARC. Des différentes expériences menées en son sein, naîtra, trois ans plus tard, le Xerox Alto qui permit alors nombre d'avancées dans le domaine des recherches interfaces homme-machine tout en posant plusieurs idées qui deviendront avec le temps des standards.

Neptune, l'explorateur de fichiers de l'Alto

Neptune, l'explorateur de fichiers de l'Alto

### **Les interactions à la souris**

Même si la souris en tant que périphérique de pointage fut développée dès 1965 par Doug Engelbart (dans l'objectif de remplacer les 'light-pens' dont l'usage constitue un standard depuis les travaux de Sutherland), c'est bien les recherches du Xerox Parc dans le cadre du projet Alto qui imposèrent son utilisation afin de permettre la manipulation directe d'objets graphiques mise au point avec le Sketchpad.

D.Engelbart tenant en main sa souris informatique

D.Engelbart tenant en main sa souris informatique

## **WIMP**

L'usage d'un dispositif de pointage abordable permet le développement de ce qui deviendra une interface W(I)MP (Windows, Icons, Menus, Pointing device, soit Fenêtres, Icônes, Menus et dispositifs de pointage). (A noter que le I pour icônes est ici entre parenthèses étant donné que ce concept sera introduit plus tardivement).

Ce type d'interface repose sur un certain nombre de principes de base tels que:

L'usage de fenêtres, qui permet l'affichage de plusieurs programmes, chaque fenêtre étant chargée d'afficher un seul et unique programme. Ce concept de fenêtre s'emplant et s'entrechevauchant ayant été présentés en 1969 par Alan Kay.

La présentation de menus, dont les diverses entrées présentent chacune une possibilité d'action offerte à l'utilisateur.

La manipulation de ces éléments à l'aide d'un dispositif de pointage (ici une souris).

Ces différents éléments sont présentés en détail plus bas.

## **WYSIWYG**

Le Xerox Alto dispose en 1974 du premier éditeur de texte graphique de type WYSIWYG ("What you see is what you get", littéralement "Ce que vous voyez est ce que vous obtenez"): Bravo. Cette vision de l'interaction (WYSIWYG) vise à permettre un rapprochement entre ce que l'utilisateur perçoit et manipule, et le rendu final qu'il obtient. Concrètement, dans le cas d'un éditeur de texte WYSIWYG, le logiciel présente à l'utilisateur une page (à la manière d'une feuille de papier), sur laquelle il peut écrire tout en voyant le résultat de sa mise en page en temps réel.

## Bravo, l'éditeur de document

Bravo, l'éditeur de document

A noter également que le Xerox Alto participe à la popularisation de l'usage des graphismes matriciels précédemment abordés.

Malgré toutes ces innovations, l'Alto ne fut pas commercialisé, Xerox ne sachant pas alors, comment exploiter sur le marché l'ensemble des avancées réalisées par son laboratoire de recherche. Ce sera le Xerox Star qui, permettra quelques années plus tard la diffusion de ces concepts.

## **Xerox Star**

C'est en effet en 1977 que le développement du Xerox Star débute, en se basant sur les résultats obtenus durant le projet Alto. Tout le projet repose sur l'idée de "bureau virtuel" conceptualisée par le Dr. David Canfield Smith. Celle-ci prend la forme d'une interface représentant à l'écran un ensemble de fichiers et de dossier, représentés par des icônes (permettant d'ajouter le I au W(I)MP présenté plus haut). L'objectif étant ici de réaliser une interface la plus *intuitive* possible, en se basant notamment sur l'idée d'interface WYSIWYG.

Coté technique, l'usage du SmallTalk et de la programmation orientée objet facilita grandement le développement du système. Chaque élément étant représenté par un objet informatique correspondant à son type. Coté utilisateur, cet objet est représenté graphiquement et l'utilisateur est en mesure d'interagir avec celui-ci à l'aide de son dispositif de pointage (ici la fameuse souris).

Le Xerox Star, présentant différentes fenêtres ainsi qu'un bureau virtuel en fond.

Le Xerox Star, présentant différentes fenêtres ainsi qu'un bureau virtuel en fond.

# **Les standards d'interaction**

Malgré des succès commerciaux tout relatifs, les différents projets conduits par le Xerox Parc poseront les bases sur lesquels reposeront l'ensemble des interfaces homme-machines graphiques qui seront produites à l'avenir, qu'il s'agisse de concepts généraux ou d'éléments d'interface à travers le concept WIMP précédemment présenté. C'est Apple, avec son célèbre Macintosh, qui popularisera à partir de 1984, cette conception de l'interaction homme-machine auprès du grand public.

## **Les actions réalisables par l'utilisateur**

L'interaction graphique repose sur un ensemble d'actions primaires réalisables sur un ensemble de composants.

Parmi ces actions de base on distingue:

### **Le pointage**

Le pointage consiste simplement à déplacer le curseur d'un dispositif de pointage sur une zone de l'interface graphique.

### **Les clics (ou pointer-et-cliquer)**

Un clic sur réalise en appuyant sur un des boutons physiques présent sur un dispositif de pointage (typiquement, une souris). Un clic sur un élément signale à la machine que l'utilisateur souhaite interagir avec un élément de l'interface.

### **Le déplacement**

Un déplacement consiste pour l'utilisateur, à réaliser un clic puis, sans relâcher la pression exercée sur le bouton à l'origine de ce clic, déplacer le dispositif de pointage.

Cette action de déplacement permettant, entre-autre, de mouvoir un élément de l'interface d'un emplacement à un autre.

## **Le survol**

Le survol consiste à placer le pointeur du dispositif de pointage au dessus d'un composant graphique. Si celui-ci gère le survol, le composant agira selon sa fonction.

## **Le raccourci**

Si les interfaces graphiques permettent de présenter à l'utilisateur un environnement de travail plus intuitif, ce gain de praticité est généralement associé à une perte de vitesse. Aussi, afin de contourner cette problématique, bon nombre d'interfaces graphiques associent aux actions qu'elles proposent des raccourcis qui consistent généralement en une combinaison de touches du clavier.

Lorsque l'utilisateur réalise une combinaison de touches associée à un raccourci, le système réalise l'action associée audit raccourci.

## **Les composants graphiques**

Comme indiqué plus haut, le panel d'action offert à l'utilisateur se réalise sur un ensemble de composant, énumérés ci-dessous, ayant chacun un rôle défini.

### **L'icône**

Une icône est un pictogramme présentant à l'utilisateur une information.

Une icône de disquette, représentant généralement l'action de sauvegarder

Une icône de disquette, représentant généralement l'action de sauvegarder

### **Le libellé**

Un libellé est un simple groupe de mots présent à l'écran. Il permet de transmettre une information à l'utilisateur en l'exprimant de manière précise à travers une expression verbale.



## La fenêtre

Une fenêtre présente un espace rectangulaire dans lequel un programme peut afficher son interface. Une fenêtre peut-être déplacée et plusieurs fenêtre peuvent se superposer.

Chaque programme dispose d'une ou plusieurs fenêtre pour interagir avec l'utilisateur.

Cet ensemble de fenêtres est géré par un superviseur appelé Gestionnaire de fenêtres (Windows Manager en anglais).

Une fenêtre est un agrégateur de composants. C'est à dire qu'il s'agit d'un composant graphique affichant lui-même un sous-ensemble d'éléments visuels.

Julien Henry, auteur original&nbsp;: Shmuel Csaba Otto Traian [[CC BY-SA 3.0](http://creativecommons.org/licenses/by-sa/3.0) ou [GFDL](http://www.gnu.org/copyleft/fdl.html)], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AFen%C3%AAtre_(syst%C3%A8me_de_fen%C3%AAtirage)_fr.svg)

Julien Henry, auteur original&nbsp;: Shmuel Csaba Otto Traian [[CC BY-SA 3.0](http://creativecommons.org/licenses/by-sa/3.0) ou [GFDL](http://www.gnu.org/copyleft/fdl.html)], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AFen%C3%AAtre_(syst%C3%A8me_de_fen%C3%AAtirage)_fr.svg)

## La barre de défilement

La barre de défilement ou ascenseur (scrollbar) est un composant d'interface graphique qui permet à l'utilisateur, par l'intermédiaire d'un déplacement de ladite barre, de naviguer au sein d'un document, ou d'une fenêtre en général.

Cette barre peut-être verticale (afin de permettre un déplacement vertical) ou horizontale (afin de permettre un déplacement horizontal).

Par Original téléversé par Marc Mongenet sur Wikipedia français [[CC BY 2.5](http://creativecommons.org/licenses/by/2.5)], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AScrollbar2.png)

Par Original téléversé par Marc Mongenet sur Wikipedia français [[CC BY 2.5](http://creativecommons.org/licenses/by/2.5)], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AScrollbar2.png)

## Le menu

Un menu est un élément d'interface graphique généralement rectangulaire chargé de présenter une liste d'options à l'utilisateur. Un menu regroupant plusieurs options présentant toutes un lien logique entre-elles.

Un menu présente principalement une simple phrase (ou mot) à l'utilisateur, illustrant le lien logique qui lie les options entre-elles. Lors du survol ou du clic sur cette phrase (ou mot), l'ensemble des options associées au menu deviennent visibles, permettant alors à l'utilisateur de réaliser diverses actions.

Par Jaksmata (Travail personnel) [Public domain], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AMenu_(computing)_example.PNG)

Par Jaksmata (Travail personnel) [Public domain], [via Wikimedia Commons](https://commons.wikimedia.org/wiki/File%3AMenu_(computing)_example.PNG)

## La palette

Une palette est une simple boîte, généralement rectangulaire, présentant une sélection d'icônes associée chacune à une action bien définie.

## Le bouton

Un bouton est une composant sur lequel il est possible de réaliser un "pointer-et-cliquer" entraînant une opération bien définie par la machine.

Le bouton porte généralement un texte ou une icône exprimant l'opération qui lui est associée.

Par Ar mythra (Travail personnel) [Public domain], <a href="https://commons.wikimedia.org/wiki/File%3ACommand\_button.png">via Wikimedia Commons</a>

Par Ar mythra (Travail personnel) [Public domain], <a href="https://commons.wikimedia.org/wiki/File%3ACommand\_button.png">via Wikimedia Commons</a>

## Le bouton radio

Le bouton radio est toujours présenté au sein d'un groupe de boutons radios. On l'utilise pour sélectionner une seule option parmi un ensemble d'options.

Un bouton radio présente deux états: coché et décoché. Lors du clic sur un bouton radio décoché, celui-ci passera à l'état "coché" tandis que tout les autres boutons radio de son groupe passeront à l'état "décoché".

Par manucoThis image is an old version created by Medium69.Cette image est une ancienne version créée par Medium69.Please credit this : William Crochot (Travail personnel) [Public domain ou <a href="http://creativecommons.org/licenses/by-sa/2.0/fr/deed.en">CC BY-SA 2.0 fr</a>], <a href="https://commons.wikimedia.org/wiki/File%3ARadioButtonsGui\_(french\_version).png">via Wikimedia Commons</a>

Par manucoThis image is an old version created by Medium69.Cette image est une ancienne version créée par Medium69.Please credit this : William Crochot (Travail personnel) [Public domain ou <a href="http://creativecommons.org/licenses/by-sa/2.0/fr/deed.en">CC BY-SA 2.0 fr</a>], <a href="https://commons.wikimedia.org/wiki/File%3ARadioButtonsGui\_(french\_version).png">via Wikimedia Commons</a>

## La case à cocher

Une case à cocher permet à l'utilisateur d'exprimer un choix. Une case à cocher associée à un libellé présente la possibilité offerte à l'utilisateur. Une case à cocher présente deux états: coché et décoché. En cliquant sur une case à cocher décochée, celle-ci sera généralement remplie par une croix ou un symbole "validé" signifiant ainsi au système le choix de l'opérateur.

En cliquant sur une case à cocher cochée, celle-ci sera vidée (laissée blanche).

Par Ar mythra (Travail personnel) [Public domain], <a href="https://commons.wikimedia.org/wiki/File%3ACheck.png">via Wikimedia Commons</a>

Par Ar mythra (Travail personnel) [Public domain], <a href="https://commons.wikimedia.org/wiki/File%3ACheck.png">via Wikimedia Commons</a>

## Le champs de texte (ou zone de texte)

Un champs de texte est une zone interactive dans laquelle l'utilisateur peut saisir du texte, si ce premier est activé.

En cliquant sur un champs de texte, l'utilisateur l'active et peut donc entrer le texte de son choix.

Plusieurs champs de textes associés à des libellés

Plusieurs champs de textes associés à des libellés

## L'environnement de bureau

L'environnement de bureau, ou métaphore du bureau est un concept d'interface homme-machine aujourd'hui hégémonique introduit par Alan Kay en 1970 pour le Xerox Alto, par la suite développé pour être intégré au Xerox Star.

L'idée est de représenter l'interface utilisateur à la manière d'un bureau (espace de travail) classique. Ainsi, des îcones sont utilisées pour représenter des fichiers informatique et des dossiers, qu'il devient possible de déplacer et de classer par pointer-et-cliquer ou déplacement.

Toutes les possibilités d'action sont visibles à l'écran et celles-ci ne nécessitent plus l'apprentissage d'un langage de commande. Cette vision de l'interface reposant sur le principe WYSIWYG exposé plus haut.

Par exemple, lors d'un pointer-et-cliquer sur un fichier, celui-ci est ouvert au sein du logiciel graphique (disposant d'une fenêtre) capable de le traiter. Il est possible de déplacer les fenêtres pour mieux les présenter, l'environnement de bureau restant systématiquement en arrière plan, visible ou non.

Le Mac OS d'Apple et sa métaphore de bureau

Le Mac OS d'Apple et sa métaphore de bureau

## **II. a.6 Bibliothèques et organisation technique de l'affichage**

La popularisation des méthodes d'interaction évoquées plus haut poussèrent la communauté scientifique à tenter de mettre au point un certain nombre de standards techniques permettant des implémentations des concepts d'interaction Homme-machine graphiques.

# Le système de fenêtrage X

Parmi ces standards, on peut relever la notable existence du système d'affichage graphique X (X Window System, appelé X11 ou simplement X) développé et proposé par le MIT (Massachusetts Institute of technology, Institut de technologie du Massachusetts) en 1984.

Avec X, le MIT propose un système de gestion d'interface graphique basé sur un modèle client-serveur, permettant notamment l'affichage distant.

En effet, lorsque une implémentation de X fonctionne sur une machine (serveur X), il devient alors possible pour un utilisateur de se connecter à X depuis un terminal, que ce soit à travers un réseau ou depuis la machine jouant le rôle de serveur X elle-même, afin de pouvoir disposer d'un environnement d'interaction graphique. Le serveur X et ses clients communiquent en utilisant le protocole X, les clients déléguant les tâches IHM au serveur X à travers leur communication.

Chaque logiciel faisant appel à un serveur X pour proposer une interface graphique est appelé "client X", et est indépendant des autres logiciels en cours d'utilisation par le manipulateur humain. X gérant les actions de l'utilisateur sur les éléments graphiques, comme illustré ci-dessous.

## Illustration du fonctionnement d'un serveur X et de ses clients

Illustration du fonctionnement d'un serveur X et de ses clients

X repose sur l'affichage matriciel précédemment présenté.

Le protocole X est détaillé au sein d'une bibliothèque (Xlib) diffusée parmi les concepteurs de logiciels, qui permet donc à ces derniers de faire communiquer leur application graphique avec le serveur X simplement. Comme son nom l'indique, Le système de fenêtrage

X permet la gestion d'une interface Homme-machine exposant des fenêtres présentées dans la section précédente. X prévoit entre-autre, l'affichage et le déplacement de ces fenêtres. Chaque logiciel client de X pouvant afficher une ou plusieurs fenêtres.

X n'étant en soi qu'un ensemble de spécifications proposés par le MIT, il existe de nombreuses implémentations de X (X.org, XFree86, Quartz etc...). Ce mode de fonctionnement permet aux développeur de programmer des applications graphiques indépendant du matériel sur lequel elles sont sensées fonctionner.

## **Les bibliothèques graphiques**

X ne gérant que l'affichage des fenêtres ainsi que les interactions avec celles-ci et non pas leur contenu, un certain nombre de bibliothèques graphiques de plus haut niveau ont été développé afin de faciliter la programmation du contenu de ces fenêtres.

On note ainsi l'apparition et le développement d'un empilement de bibliothèques permettant une abstraction et une uniformisation des interfaces toujours plus grandes.

Parmi celles-ci, il est possible de relever l'ensemble cairo, GTK+ fonctionnant avec X, utilisé entre autre dans le cadre du projet GNOME, proposant un environnement de bureau. Cairo étant chargé de réaliser les dessins en coordination avec X, et GTK+ spécifiant l'apparence des éléments d'interfaces (barres de défilements, boutons etc...) à présenter sur les fenêtres, reposant sur Cairo.

### Illustration de la construction GTK+, Cairo

Illustration de la construction GTK+, Cairo

Il existe bien sûr d'autres constructions de ce type mais les énumérer toutes ne présente pas spécialement d'intérêt.

# **II.b Prendre de la hauteur**

## **II.b 1 - Considérations et conséquences relatives à ces évolutions**

D'abord conçu pour répondre à des besoins liés à des usages militaires et scientifiques, l'informatique intéresse de plus en plus à l'aube des années 70-80 les acteurs économiques qui perçoivent les bénéfices dont l'économie peut tirer d'un usage de l'informatique.

En effet, à partir de 1960 et suite aux travaux de Dijkstra, Wirth et d'autres chercheurs, les liens entre informatique et mathématiques sont fortement développés, entraînant une progression des possibilités de développement, en permettant une plus grande complexité algorithmique (notamment à travers l'amélioration des langages de programmation).

Ces éléments, associés à des évolutions matérielles (miniaturisation des composants, réduction des coûts), permettent l'application de l'informatique à un plus grand nombre de secteurs (scientifiques, commercial, domestiques).

De fait, l'informatique n'est plus seulement un problème rattaché au monde de la recherche scientifique et de son public d'initiés, mais une préoccupation plus globale.

Aussi, les problématiques liées à la conception d'interfaces homme-machine se développent rapidement.



En effet, il faut rappeler qu'une interface homme-machine prend concrètement la forme du dispositif qui permet à l'utilisateur d'interagir avec l'ordinateur.

Or, avec le changement d'environnement d'opération de la machine (passage du monde militaire/scientifique au monde de l'entreprise), la manière d'utiliser la machine doit évoluer à plusieurs niveaux:

- Le public auquel s'adresse la machine change. On passe en effet d'un milieu où l'utilisateur est un initié, qui grâce à un apprentissage plus ou moins important sait comment manipuler celle-ci, à un environnement dans lequel la manipulation de la machine est un moyen, sensé être plus productif, de remplir une mission, d'atteindre un objectif.
- Les tâches que doit effectuer la machine changent également. Se contenter de réaliser des calculs et de présenter des résultats n'est plus suffisant. L'essentiel des travaux concernera le traitement de document, le stockage et le traitement d'information.

Comme annoncé plus haut, cette évolution et cette diversification des usages est intimement liée aux évolutions techniques et matérielles. En effet, on se trouve dans une situation de cercle vertueux. Le progrès technique permet une progression des possibilités d'usage, qui attise l'intérêt du monde du privé (entreprise, particuliers) qui encourage la production. Cette hausse de la production va entraîner une baisse de coûts des technologies utilisées (par effet d'économie d'échelle). Baisse des coûts qui entraîne le développement d'une meilleure accessibilité financière.

Cette accessibilité financière, n'aurait pas eu d'effets significatifs si elle n'avait pas été accompagnée d'un développement des interfaces homme-machine.

En effet, comment rendre le produit attrayant pour l'investisseur/utilisateur s'il n'est en mesure de l'utiliser sans une formation longue et coûteuse ?

(L'histoire et l'échec du Lisa d'Apple, porté par Steeve Jobs illustre parfaitement ce souci de concevoir une interface attrayante.)

Cette progression de l'importance d'être en mesure de présenter à l'utilisateur l'information de manière claire et [attrayante] soulève un certain nombre de questions.

En effet, il devient nécessaire de présenter une interface cohérente et claire à l'utilisateur et donc, de la réfléchir.

## **II.b 2 - Penser son interface**

Cette nécessité de réfléchir son interface n'apparaît cependant brusquement, tandis que la demande se diversifie.

En effet, on peut noter l'existence du projet NLS/Augment, mené par Douglas Engelbart au début des années 60.

L'objectif de ce projet était de développer une machine (équipée d'une souris et d'une gestion de fenêtres) permettant de travailler sur de l'hypertexte.

Fait intéressant, Engelbart basa la conception de cette interface sur la manière dont les enfants apprennent. Ce choix finit par aboutir à la conception d'une interface construite autour d'un mécanisme de coordination oeil-main, réellement plus simple à appréhender que l'habituel ensemble de commandes à mémoriser pour être en mesure d'interagir avec une machine.

On assiste ainsi ici, au développement d'une interface centrée sur l'accessibilité. L'apprentissage préalable nécessaire à la manipulation de la machine étant de fait moindre (à l'inverse des habituels langages de commandes). Cet objectif ayant ici guidé la conception de l'entièreté du projet.

Si Engelbart a fait le choix de l'accessibilité, ce n'est en aucun cas le seul élément à prendre en compte dans la réflexion préalable à la conception d'une interface.

Par exemple, dans certaines situations, la rapidité d'utilisation peut-être plus importante que l'accessibilité. Par exemple, un langage de commande peut présenter l'avantage de permettre une grande rapidité d'utilisation, une fois les commandes mémorisées, à l'inverse d'une interface graphique basée sur le pointage de zones de l'écran.

Lors de la conception de l'interface il est donc nécessaire de réaliser un certain nombre de choix tournant notamment autour de l'ensemble efficacité/accessibilité/rapidité d'utilisation, et coûts de développement et de recherche.

## **II.b 3 Accessibilité et design universel**

Plus haut, nous avons présenté l'idée d'accessibilité, caractéristique ayant guidé les travaux de D. Engelbart sur le NLS/Augment. Nous allons la développer un peu plus ci-dessous.

Le gouvernement Français (dans le cadre de la loi sur le handicap du 11 février 2005) propose une définition de l'accessibilité parfaitement applicable dans notre cas: L'accessibilité requiert la mise en œuvre des éléments complémentaires, nécessaires à toute

personne en incapacité permanente ou temporaire pour [...] accéder librement et en sécurité au cadre de vie ainsi qu'à tous les [...] services, produits et activités.

Ainsi et spécialement dans le cadre de la conception d'interface, l'accessibilité d'un service se mesure à la facilité d'utilisation de celle-ci par les personnes les plus démunies. Le terme d'incapacité permanente ou temporaire correspondant ici, entre autre, à la non-connaissance de l'utilisateur de l'interface. Il s'agit donc ici, en vue de rendre l'interface accessible de simplifier sa courbe d'apprentissage. C'est à dire de jouer sur l'intuition de l'utilisateur, l'interface devant être conçue de sorte que l'utilisateur prennent rapidement (immédiatement) conscience de l'ensemble des fonctionnalités que lui propose le logiciel et de comment il peut y accéder.

Néanmoins cette problématique de la facilité d'apprentissage, de l'aspect intuitif d'une interface n'est pas le seul à prendre en compte lorsque on la produit. En effet, comme le précise la description au sens large exposée plus haut, l'incapacité de l'utilisateur peut ne pas se limiter à des problématiques de connaissances, mais bien à des handicaps.

Il ne faut pas perdre de vue qu'une interface, en l'état actuel des choses, nécessite la mobilisation des cinq sens de l'utilisateur ainsi que d'une certaine capacité de réflexion. De fait, si ce dernier présente une déficience physique ou mentale il se peut qu'il ne puisse utiliser l'interface dans des conditions normales, satisfaisante. Une interface accessible devra donc veiller à proposer à l'utilisateur des options lui permettant tout de même d'interagir avec le système et ce malgré son handicap.

A noter que l'organisme de normalisation international ISO dispose d'un groupe de travail chargé de proposer des normes relatives à l'accessibilité des interfaces. Ainsi nous pouvons citer entre autre:

- La norme ISO/IEC TS 20071-11:2012, décrivant des lignes directrices relatives aux texte alternatif à présenter à la place ds images pour les personnes mal-voyantes. <https://www.iso.org/fr/standard/59423.html?browse=tc>
- La norme ISO/IEC TS 20071-25:2017, relative à la présentation audio de texte dans les vidéos.
- Le comité ISO/TC 173, chargé de produire des normes (87 à ce jour) concernant les appareils et accessoires fonctionnels pour les personnes handicapées

Dans les faits, une interfaces présente différents niveaux d'accessibilité en fonction des possibilités d'interaction qu'elle offre à l'utilisateur handicapé (au sens commun) ou non.

De manière plus générale, cette idée de concevoir des interfaces accessibles pour le plus grand nombre s'insrit dans le cadre d'une logique de conception (design) universel qui justement se définit par "La conception de tout [...] programme ou service qui puisse être utilisé par toute personne, sans nécessiter ni d'adaptation [...] spéciale, et ce quel que soient son sexe, son âge, sa situation ou son handicap."

Cette façon de penser son interface fut primairement exprimée par l'architecte Ronald L. Mace puis théorisée par Selwyn Goldsmith, dans son livre *Designing for the Disabled* (concevoir pour les personnes à capacité réduite) paru en 1963.

La conception universelle repose sur un certain nombre de principes qui sont:

- L'usage équitable : tout le monde peut utiliser l'interface)
- La flexibilité: L'interface s'adapte à son utilisateur
- La simplicité
- L'information doit être perceptible par tous.
- La tolérance à l'erreur: En cas d'erreur, le programme doit guider l'utilisateur
- Ne doit pas nécessiter un effort intense à l'usage, physique ou intellectuel

Néanmoins, il ne faut pas perdre de vue que l'accessibilité n'est pas le seul critère à réfléchir lors de la conception d'une interface. En effet, même si permettre un usage simple et facile pour tous est important, il ne faut pas perdre de vue que l'utilisateur interagit avec la machine dans l'idée de remplir une mission, d'atteindre un objectif et que bien souvent, il souhaite réaliser sa tâche de la manière la plus productive qui soit.

## **II.b 4 Efficacité et productivité**

L'efficacité est définie par le projet Wiktionnaire comme étant "la capacité de produire un résultat" [ref \(https://en.wikipedia.org/wiki/Universal\\_design\)](https://en.wikipedia.org/wiki/Universal_design). En effet, au delà de son accessibilité, l'interface homme-machine est avant tout là pour faire lien entre le manipulateur et la machine, que ce premier utilise en vue de remplir une tâche.

De fait, une interface efficace est simplement une interface qui permet à l'utilisateur de produire un résultat, d'effectuer la tâche pour laquelle il utilise un système informatique. En vue de cette considération, l'accessibilité d'une interface semble être un bon indicateur pour mesurer l'efficacité de celle-ci. Néanmoins ce n'est pas le seul.

En effet, principalement l'élément pouvant ralentir l'utilisateur lambda est la complexité de l'interface. Son accessibilité joue donc un rôle primordial dans un premier temps. Cependant, avec le temps l'utilisateur va acquérir une certaine expérience de l'outil logiciel

qu'il utilise, et donc de son interface. Il va donc chercher à gagner en productivité, à réaliser un maximum de tâche dans un temps donné. Une interface conçue pour être accessible (ne nécessitant aucun apprentissage particulier) nécessitera en général un temps plus important pour atteindre un objectif donné.

Un bon exemple est celui des raccourcis-claviers, qui sont généralement la combinaison d'une ou plusieurs touches, associés à une action dans un programme. Prenons la cas d'un editeur de texte WYSIWYG, l'utilisateur souhaite réaliser l'action "sélectionner tout le texte". De manière standard cette action se réalise en pointant le début du texte, puis à réaliser une action de déplacement de la souris sur tout le texte, ce qui peut se révéler long et contraignant pour l'utilisateur. Lorsque le logiciel propose un raccourcis-clavier "Sélectionner tout le texte" (généralement Ctrl+A), l'utilisateur peut l'effectuer et l'ensemble du texte est sélectionné instantanément, permettant par là un gain de temps proportionnel à la longueur du texte. Si l'action "sélectionner tout le texte" avec le dispositif de pointage est intuitive, elle n'est en revanche absolument pas productive, l'efficacité de l'interface ne proposant pas de raccourcis équivalent est donc limitée pour un utilisateur chevronné.

De fait pour être totalement efficace, l'interace ne doit pas limiter un utilisateur avertit, ayant appris à maitriser l'outil sur lequel il travaille.

Les résultats de l'étude de cas "Apprentissage assisté par ordinateur et interaction homme-machine" menée par L.-O. Pochon et Michèle Grossen apportent un certain nombre de consatations intéressantes quant à ces éléments.

Dans leur étude, M. Pochon et Mme. Grossen définissent l'ensemble des possiblités connues et maitrïsées par l'utilisateur comme étant "l'espace d'interaction" de celui-ci sur le logiciel. Un effet indéniable du temps passé par l'utilisateur à manipuler l'outil (l'interface homme-machine) est l'augmentation de son espace d'interaction. En

effet avec le temps, l'utilisateur prend non seulement conscience de l'ensemble des possibilités proposées par le logiciel mais après même à terme à contourner les règles posées par le logiciel, entre autre, en le détournant de ses fonctionnalités originelles pour en créer de nouvelles non-prévues par les concepteurs, ou encore en apprenant à contourner les messages d'erreurs, afin d'éviter leur caractère désagréable (le message d'erreur peut apparaître comme une remontrance que la machine fait à l'utilisateur) ou simplement pour gagner du temps.



## **II.b 5 Efficience et cognitive**