

Cours 7b:

ergonomie des interfaces

(règles d'usabilité)

Anastasia.Bezarianos@Iri.fr

(plusieurs slides sont basés sur des slides de N. Roussel, O. Chapius, T. Tsandilas, W. Mackay, M. Beaudouin Lafon, et S. Greenberg)

ergonomie des interfaces (utilisabilité, usabilité)

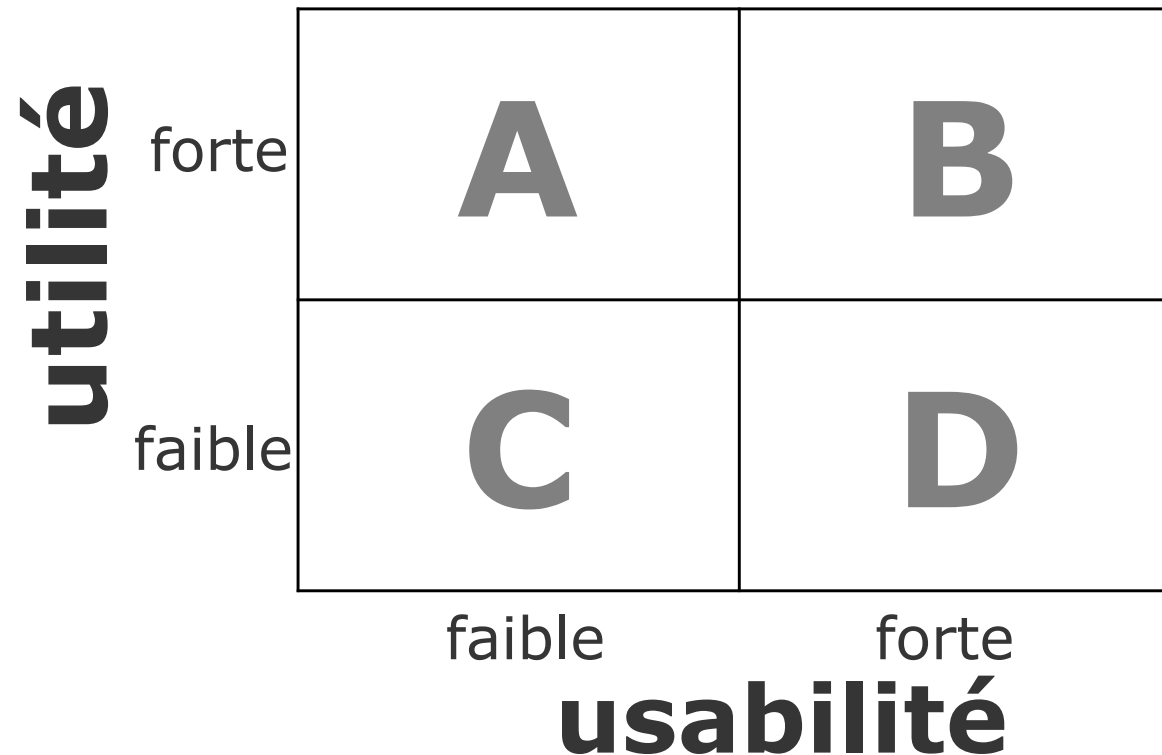
« Le degré selon lequel un produit peut être utilisé, par des utilisateurs identifiés, avec efficacité, efficience et satisfaction, dans un contexte d'utilisation spécifié.

(Norme ISO 9241)

Un système ergonomique est : facile à apprendre et mémoriser, efficace, visuellement agréable et rapide à reprendre des erreurs.

utilité

Atteindre des besoins spécifiques et soutenir des tâches réelles



D est meilleur que A ? Que pensez-vous ?

Evaluation Heuristique - Nielsen (1990)



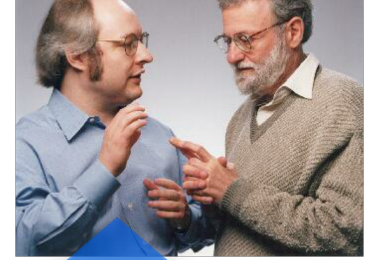
L'inspection systématique d'interface, en utilisant les catégories d'usabilité (en suite)

Procédure

- 3-5 inspecteurs (experts d'usabilité, les utilisateurs finaux)
- Inspecter l'interface (environ 1-2 heures pour les interfaces simples)
- Comparer leurs notes après

Marche pour storyboards, prototypes, et vrais systèmes

Evaluation Heuristique - Nielsen (1990)



Non guidée

- Exploration du système, sans tâche dirigée
- Aide à explorer les divers aspects de l'interface

Guidée par des Scénarios (de phase 1 ou 2)

- Utiliser des tâches représentatives des utilisateurs
- Problèmes identifiés dans les parties pertinentes
- Fonctionnalités d'intérêt sont évaluées
- Mais des problèmes peuvent échapper

Avant de commencer

Source des critères et recommandations

De l'ergonomie du logiciel au design des sites web,

Jean-François Nogier, Dunod, 2002

Ten Usability Heuristics, <http://www.useit.com/>, Jakob Nielsen

Avantages des heuristiques

Ils permettent de corriger de nombreux problèmes d'utilisabilité

Ils sont faciles à retenir, et assez simples à appliquer

Ils peuvent servir à une évaluation simple et rapide d'un système

Inconvénients

Ce n'est pas une check-list

Il y a quelques redondances

Certains critères sont à manier avec précaution

Chaque règle peut avoir des exceptions

Catégories (heuristiques)

- 1- Compatibilité
- 2- Guidage
- 3- Homogénéité
- 4- Flexibilité
- 5- Traitement des erreurs
- 6- Concision

Compatibilité

Compatibilité

Capacité du système à s'intégrer dans l'activité des utilisateurs

Recommandations :

- parler le langage de l'utilisateur
ex. vocabulaire
- présenter les informations de façon cohérente par rapport aux autres supports de travail
ex. version électronique d'un formulaire papier
- l'accès aux commandes doit être compatible avec la tâche de l'utilisateur
ex. commandes fréquentes au premier niveau des menus

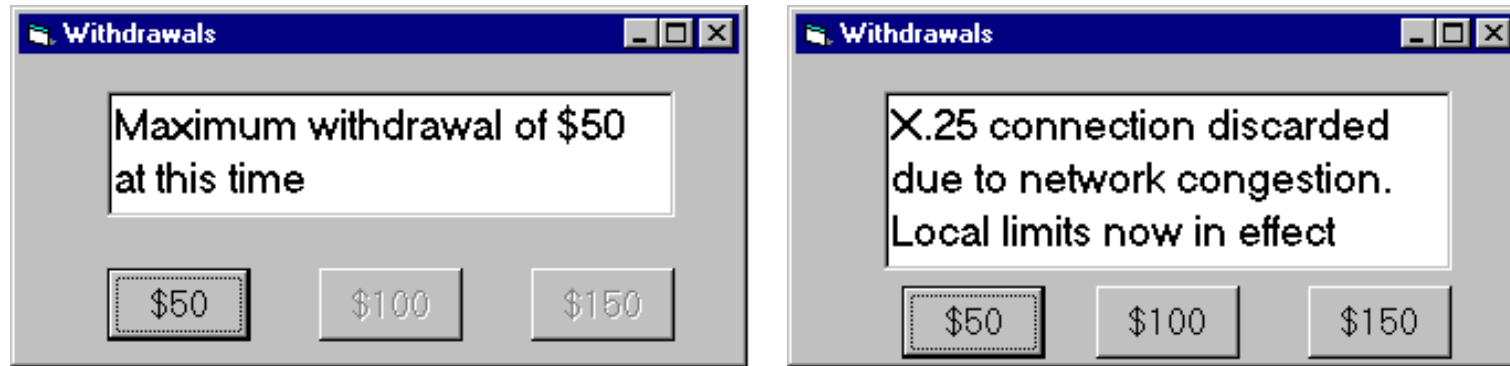
Nécessité d'étudier et d'analyser les pratiques de travail des utilisateurs visés

(parfois liée au règle de conception: métaphore)

Compatibilité (suite)

Parler le langage de l'utilisateur

ex. retirer de l'argent d'un guichet automatique

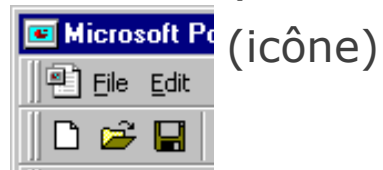


Utilisez raccourcis et icônes faciles à mémoriser

ex. Enregistrer Fichier (File) / Enregistrer (Save)

Alt+F+S (accès au menus)

Ctrl + S (raccourcis)



Compatibilité (suite)

Présenter les informations de façon cohérente par rapport aux autres supports de travail

Cheap Shop Catalog Store
Danderly software, screen A1.1

Purchaser

Name: Phone:

Postal Code: Province: City:

Delivery Address:

Today's date:

Credit Card No.: for dept use: validation id:

Catalog Item

Number: Quantity: Cost/item: Total:

Balance Owing:

Next Catalog Item (PF5)

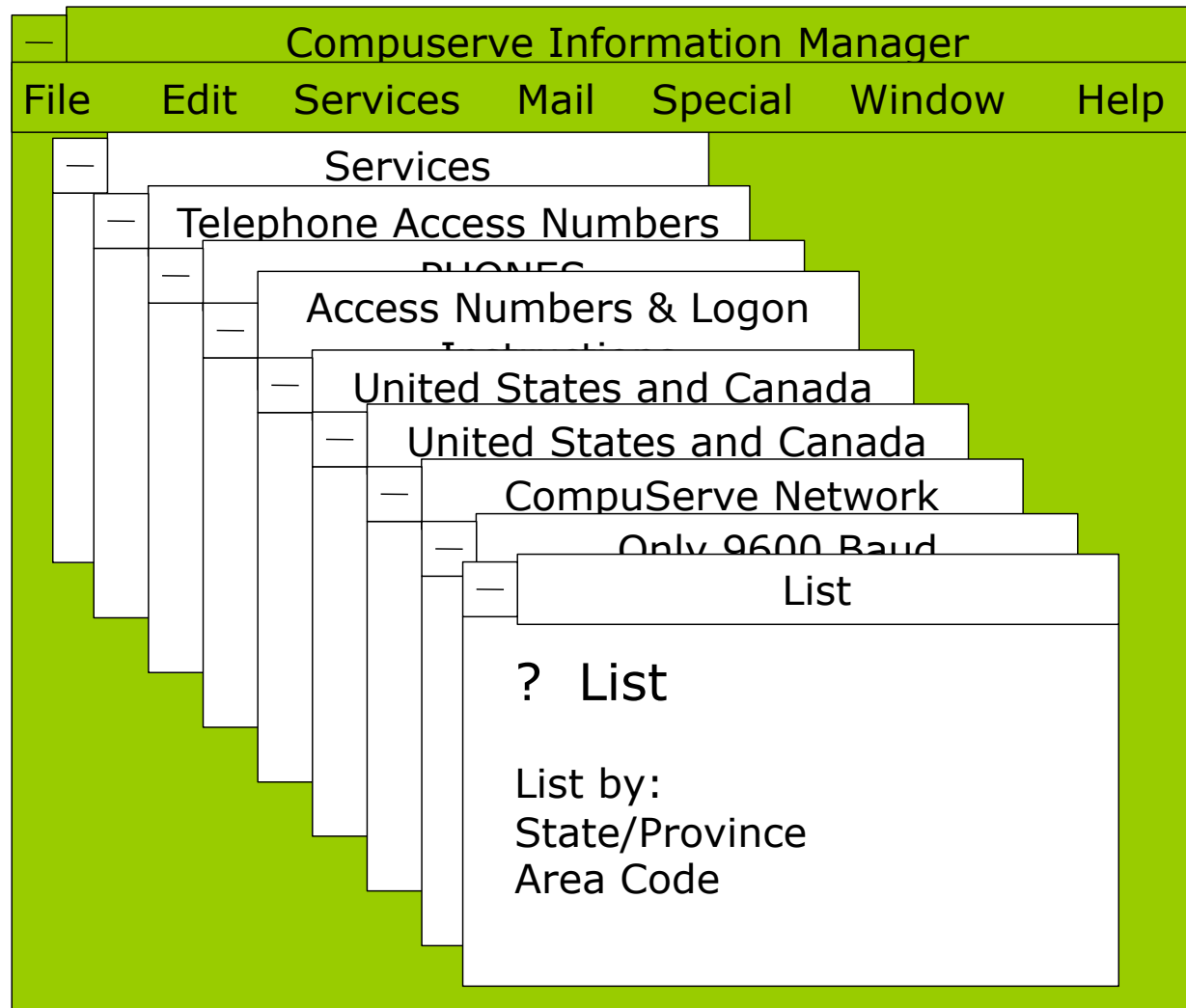
Trigger Invoice (PF8)



???

Compatibilité (fin)

L'accès aux commandes doit être compatible avec la tâche de l'utilisateur



Guidage

Guidage

Objectif : faciliter l'utilisation du système et son apprentissage

Mécanismes de retour (*feed-back* et *feed-forward*) pour

- réduire la charge cognitive
- prévenir des situations d'erreur
- rassurer (ex : progression d'une opération)

(parfois liée aux règles de conception: affordances et mapping)

Guidage (suite)

Recommandations : incitation (feed-forward)

- griser les commandes non disponibles
- fournir la liste des saisies attendues
- donner le format de saisie des données
- donner des valeurs par défaut « intelligentes »

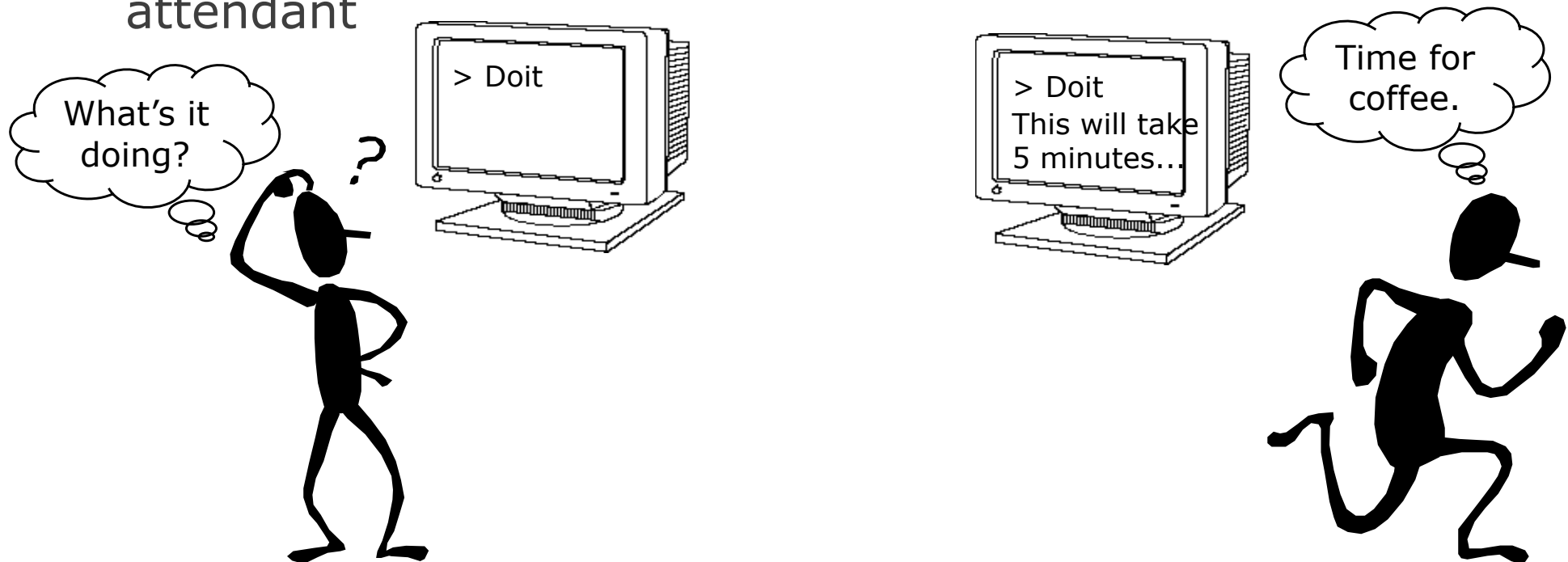
Recommandations : retour utilisateur (feed-back)

- à toute action de l'utilisateur doit répondre un changement de présentation de l'interface
- signaler les traitements longs par un message d'attente
- indiquer les modes de fonctionnement du système
- toujours faire apparaître les saisies de l'utilisateur
- rendre visibles les traitements réalisés par le système

Guidage (suite)

Traitements longs : Quelques données sur la perception du temps de réponse :

- $t < 0.1$ sec : perçu comme instantané
- $t < 1$ sec : le délai est perçu, mais ne perturbe pas l'utilisateur
- $t > 10$ secondes : l'utilisateur va vouloir faire autre chose en attendant



Guidage (suite)

Pour les délais longs :

Pour les transactions court

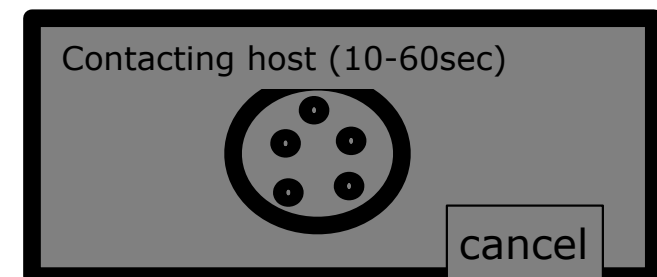
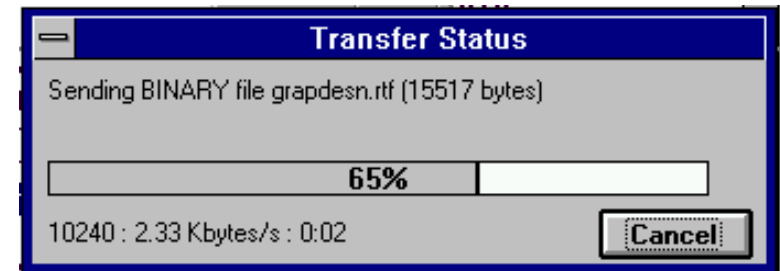
- changement de curseur

Pour les taches déterminées

- temps qui reste
- temps estimé
- avec barre de progression

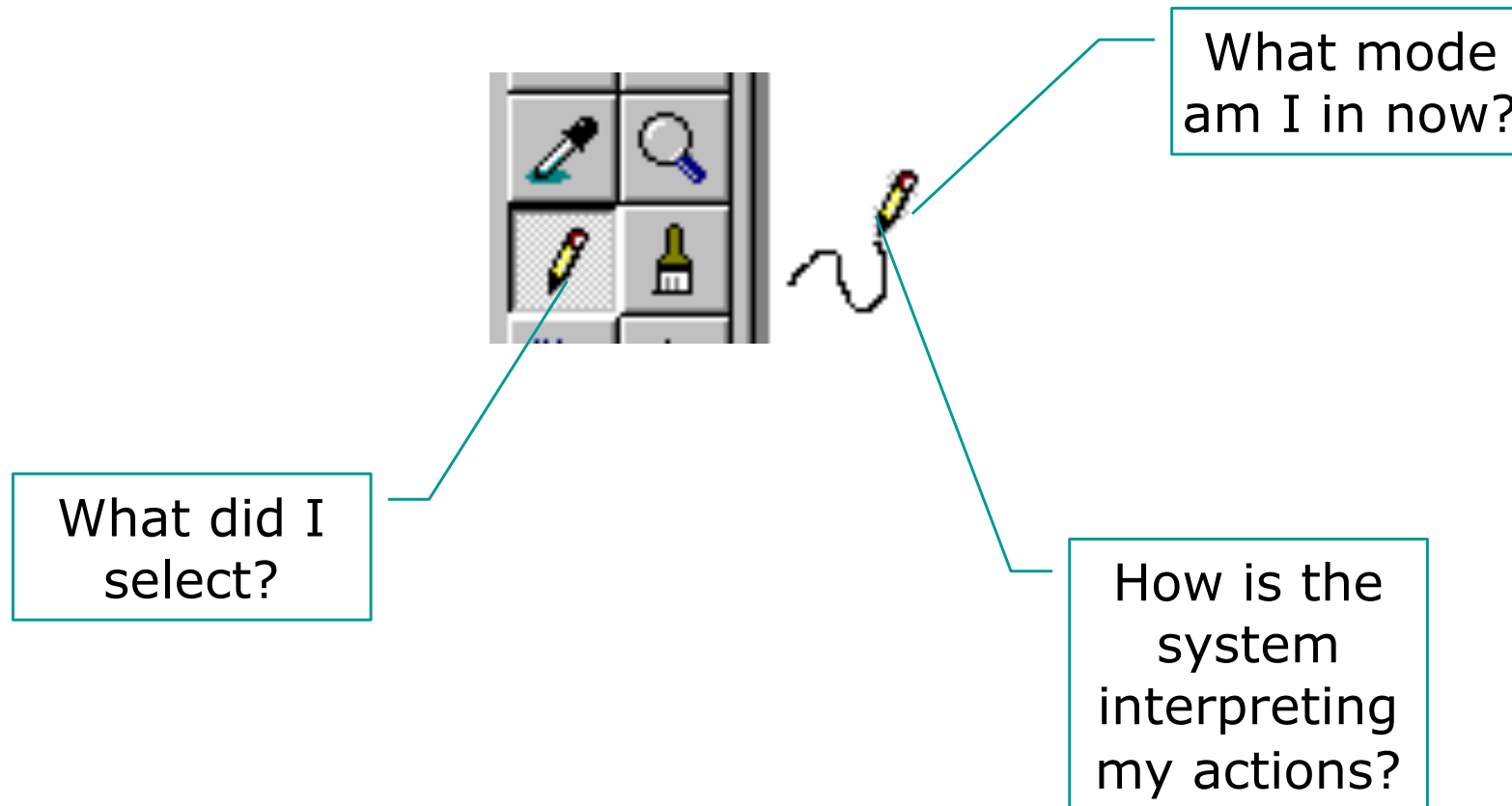
Pour des temps inconnus

- animation aléatoire



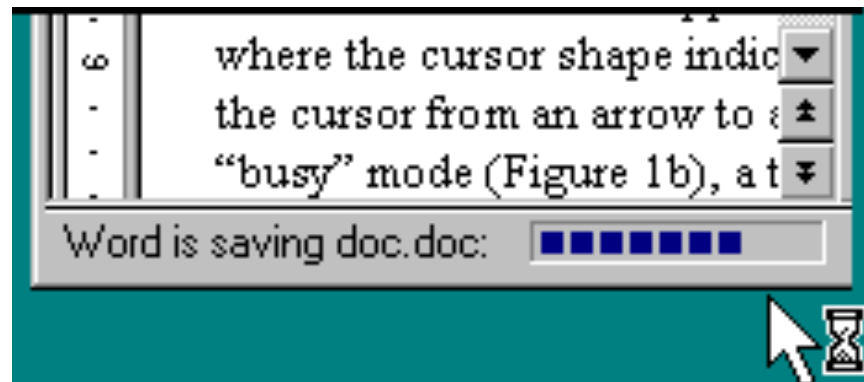
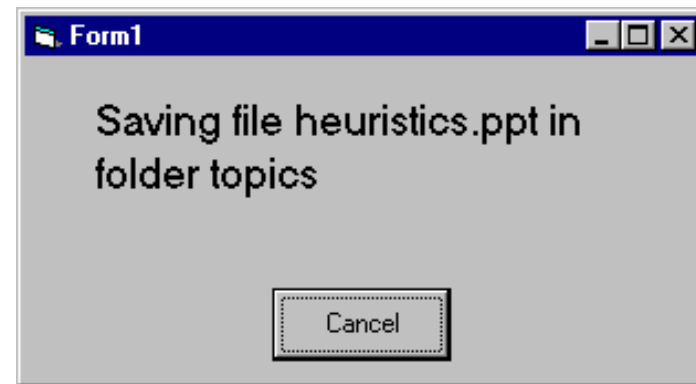
Guidage (suite)

Modes de fonctionnement



Guidage (suite)

Guidage spécifique est mieux

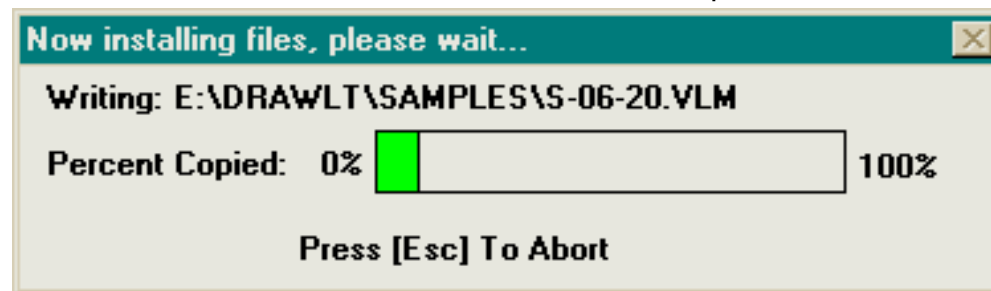


encore mieux en contexte de la tâche

Guidage (suite)

Guidage problématique

Plusieurs fichiers sont copiés, mais la barre de progression est fichier par fichier.



Drawing Board LT

Guidage (fin)

Recommandations :

groupement/distinction (design graphique)

- regrouper les informations de même type par le format ou la position
- distinguer par une présentation différente les informations distinctes

Recommandations :

lisibilité (typographie)

- utiliser une police droite
- employer des lettres sombres sur fond clair

(liée au Gestalt, psychologie, guide CRAP)

Homogénéité

Homogénéité

Concerne la cohérence globale de l'interface

- interne : à l'intérieur d'une application
- externe : entre les applications, avec la métaphore du système

Principe : un système qui a l'air familier est perçu comme simple à utiliser

Intérêt : faciliter l'apprentissage et l'utilisation

Risque : freiner ou bloquer l'évolution
(inertie des standards)

(parfois liée aux règles de conception: affordances et mapping, aussi que Consistance du CRAP)

Homogénéité (suite)

Recommandations

- les fenêtres doivent suivre le même schéma d'agencement
exemple : boîte de recherche en haut à droite
- la sémantique des boutons de la souris doit être constante
- le même vocabulaire doit être utilisé pour désigner les commandes du logiciel
exemple : ouvrir / copier-coller / préférences / ...
- la syntaxe des commandes doit être cohérente sur l'ensemble de l'interface

La consistance ne concerne pas que l'apparence visuelle
Autres exemples : syntaxe et effets des commandes

Homogénéité (suite)

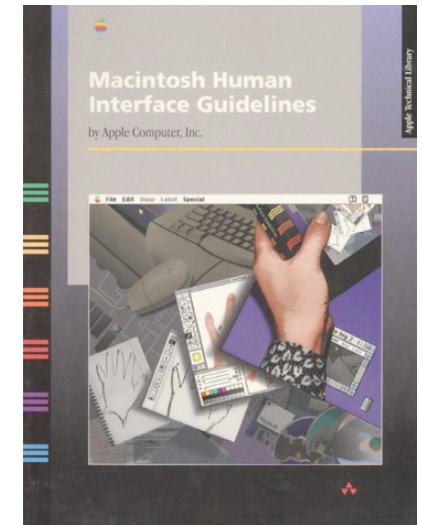
Guides de style :

- publiés par les concepteurs d'interfaces graphiques
- décrivent le "look and feel" d'une plateforme
- sont souvent vicieux : faciliter la vie de ceux qui les suivent et compliquer celle de ceux qui ne les suivent pas...

Exemples :

- OSF Motif
- IBM CUA
- Apple Human Interface Guidelines
- MS Windows

Bons sur le principe, mais peuvent être difficiles à suivre
Implémentés (en partie) dans les boîtes à outils d'interface



Homogénéité (suite)

Motif style guide v1.1 : MessageDialogs should be used to convey a message to the user. They must not interrupt the user's interaction with the application. They should include a message, and one of the following button arrangements.

OK

OK Help

OK Cancel

OK Cancel Help

Cancel

Cancel Help

Yes No

Yes No Help

Yes No Cancel

Yes No Cancel Help

Retry Cancel

Retry Cancel Help

Homogénéité (suite)

Macintosh Human Interface Guidelines

Figure 5-11 Examples of correct and incorrect window titles



Table 10-1 Pointers

Pointer	Name	Used for
	Arrow	Scroll bar, other controls, size box, title bar, close box, zoom box, menu bar, desktop
	Crosshairs	Drawing, shrinking, or stretching graphic objects
	I-beam	Selecting and inserting text
	Plus sign	Selecting fields in an array
	Wristwatch	Showing that a lengthy operation is in progress

Homogénéité (fin)

Homogénéité (et guidage) problématique

Ce sont des étiquettes avec une apparence surélevée (affordance)

Est-ce une surprise qu'on essaie de cliquer dessus?

Subscriber

Name: **Tech. Re**

Account #: **Status:**

Contact

Telephone: **E-Mail:**

Address: **St**

Save **Cancel**

Flexibilité

Flexibilité

Capacité de l'interface à s'adapter à différents contextes d'utilisation

Recommandations :

- permettre l'activation des commandes au clavier ou à la souris
- autoriser le déclenchement d'une commande fréquente depuis plusieurs fenêtres de l'application
- permettre à l'utilisateur de paramétrer le logiciel selon ses préférences
- fournir un moyen rapide d'accéder aux commandes des menus

Peut être contradictoire avec la concision (plus tard)

Flexibilité (suite)

Les utilisateurs expérimentés/experts doivent pouvoir se servir rapidement des opérations les plus courantes

Solutions :

- icônes
- abbréviations / raccourcis (Ctrl-S)
- mnémoniques (Alt-F-S)
- touches spéciales (Xerox Star: move, copy, properties...)
- complétion automatique, frappe anticipée (*type ahead*)
- mécanismes de navigation, historiques
 - WWW: ~ 60% des pages sont des revisites
- redo

Flexibilité (suite)

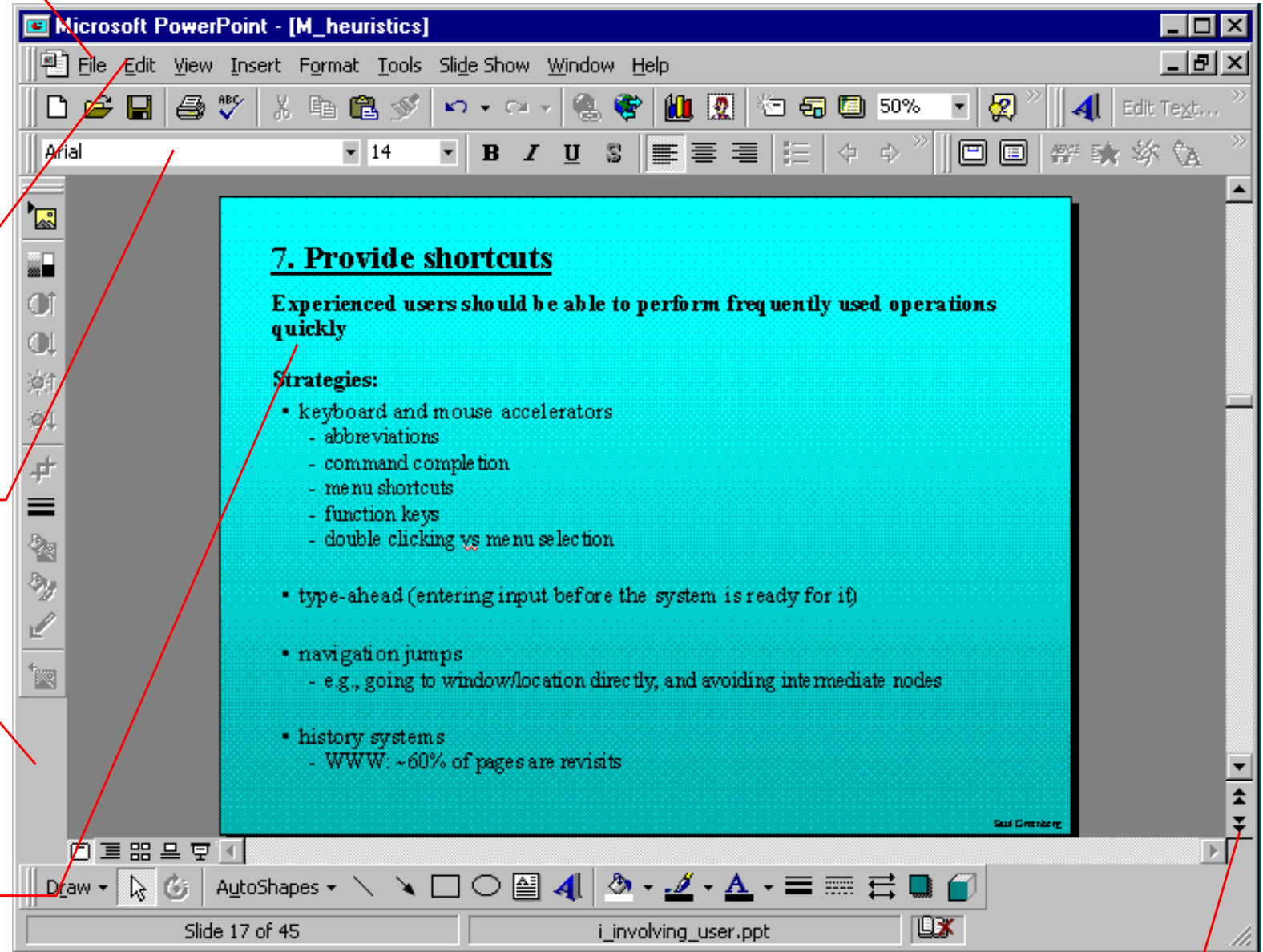
Keyboard accelerators for menus

Customizable toolbars and palettes for frequent actions

Split menu, with recently used fonts on top

Double-click raises toolbar dialog box

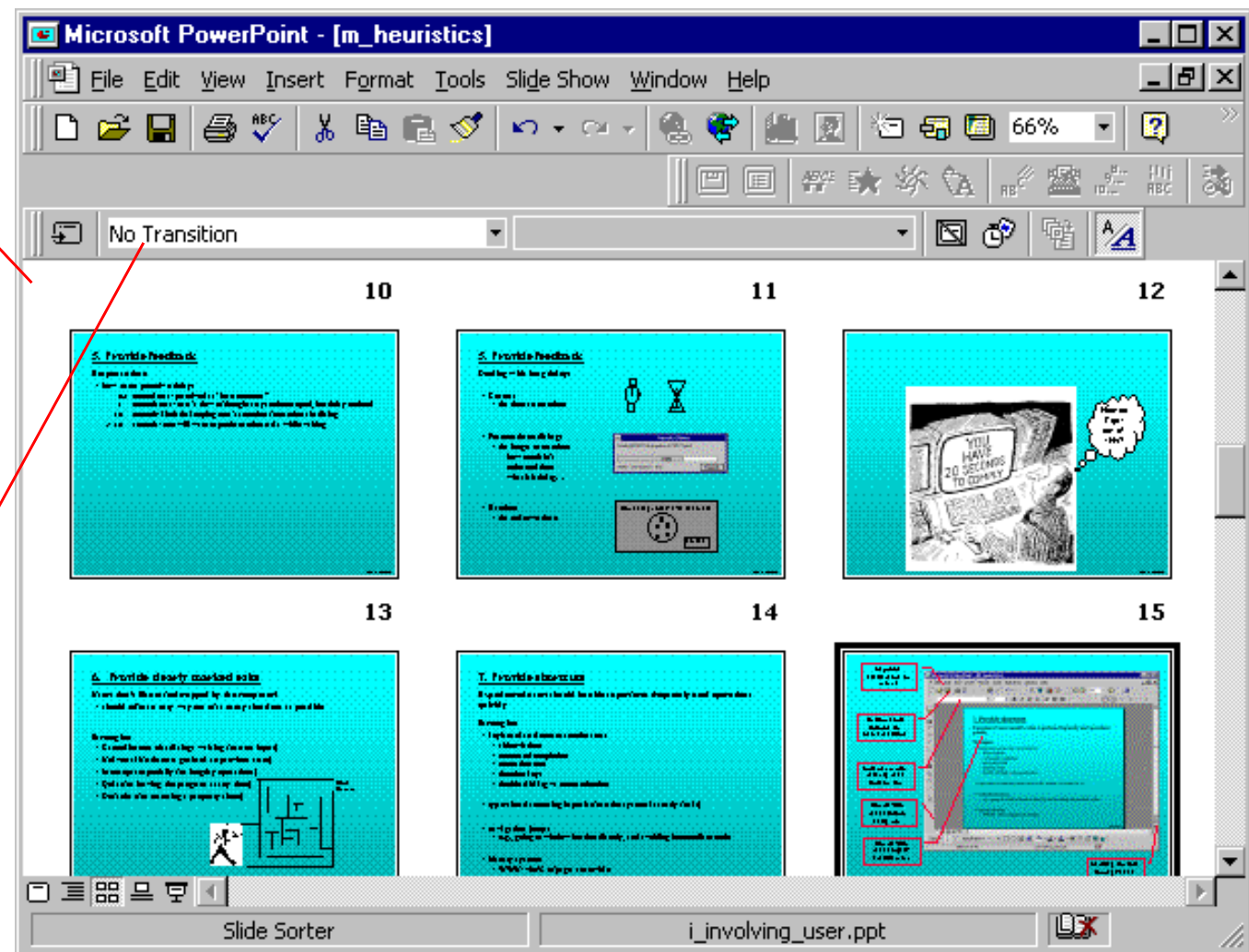
Double-click raises object-specific menu



Flexibilité (suite)

Alternate representation for quickly doing different set of tasks

Toolset brought in appropriate to this representation



Flexibilité (fin)

Adaptabilité : personnalisation explicite par l'utilisateur

- lexique
- préférences de présentation

Attention à l'interface de configuration des préférences :
d'autant plus difficile que l'application est générale

Attention à ne pas transformer l'utilisateur en concepteur

Adaptativité : personnalisation dynamique sans action explicite de l'utilisateur

L'adaptativité est très controversée...

- non prédictibilité
- rupture du principe de cohérence

Traitement des erreurs

Traitement des erreurs

Regroupe les différents moyens visant à protéger l'utilisateur des erreurs et à corriger celles-ci

Recommandations générales :

- éviter les erreurs (!)
- détecter les erreurs au plus tôt
- permettre une correction aisée des erreurs
- faciliter l'exploration et l'apprentissage du système

(parfois liée aux règles de conception: affordances, mapping, contraintes et les guides de externalisation de cognition)

Traitement des erreurs (suite)

Les utilisateurs vont faire des erreurs (c'est certain!)

Exemples :

- *habitude* : créer un nouveau document à partir d'un document existant, et l'enregistrer sans lui donner un nouveau nom
- *similarité* : l'action ressemble à d'autres qui sont possibles
- *fausse manipulation* : cliquer sur la commande voisine de celle que l'on vise
- *inattention* : oublier notre objectif en train de faire une séquence d'actions
- *incompréhension* : croire que l'on déplace une copie d'un objet alors qu'on déplace l'objet
- *exploration* : essayer un bouton que l'on ne connaît pas

Erreurs causées par actions intentionnelles et non-intentionnelles

Traitement des erreurs (suite)

On peut tirer partie de la formation des habitudes :

- la répétition et l'entraînement développent des habitudes
- ceci est bien pour la performance

Réduire le nombre de choix peut favoriser le développement d'habitudes

Mais ... les habitudes sont sources d'erreurs :

ex. valider sans réfléchir une boîte de confirmation

Traitement des erreurs (suite)

Solutions (entre autres) :

- empêcher les manipulations qui produisent des erreurs
- faciliter la détection et la correction d'erreur par
 - le feed-back
 - un historique (*défaire* - *refaire*)

A screenshot of a window titled 'Form1'. It contains a 'Date:' label followed by a single text input field. Below this, there are three separate input fields for 'Month', 'Day', and 'Year'. Further down, there is a date picker showing 'May 22 1997' with dropdown arrows for each component. At the bottom, there is another date picker with the same date, but with small square buttons next to each component for selection.

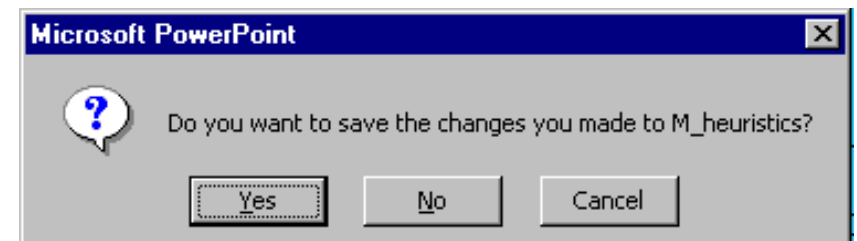
A screenshot of a window titled 'Appointment'. It has tabs for 'General', 'Attendees', 'Notes', and 'Planner'. The 'General' tab is active. It shows 'When' section with 'Start' and 'End' time and date pickers. Below this is a 'Description' text area. A calendar for 'May 1997' is displayed, showing the date '14' (Wednesday) selected. At the bottom, there is a 'Where:' label and a text input field.

A screenshot of a 'History' and 'Macros' panel. It has two tabs: 'History' and 'Macros'. The 'History' tab is active. It shows a list of actions performed, including 'EnabledNewTagIcon.gif', 'Open', 'Eraser', 'Eraser', 'Pencil', 'Pencil', 'RGB colors', and 'New Layer'. The 'Pencil' action is currently selected. The panel has a scroll bar on the right and a toolbar at the bottom with icons for undo, redo, and delete.

Traitement des erreurs (suite)

Recommandations : protection contre les erreurs

- feed-forward:
 - griser les commandes non disponibles
 - fournir la liste des valeurs possibles, et de bonnes valeurs par défaut
- minimiser les saisies clavier
- détecter les erreurs au plus tôt
 - faire des « checks » des valeurs raisonnables
- limiter les pertes de données (sauvegarde automatique, journal)
- valider explicitement les commandes irréversibles



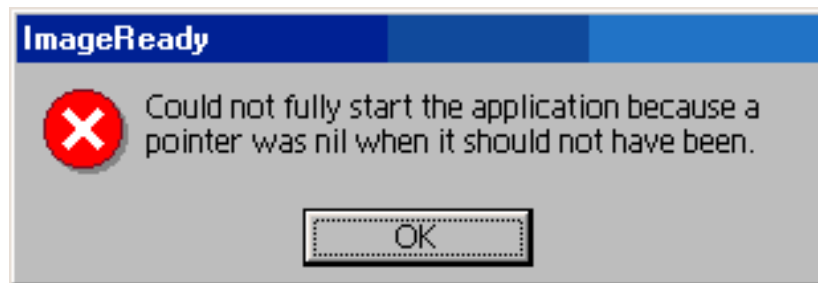
Traitement des erreurs (suite)

Recommandations : correction des erreurs

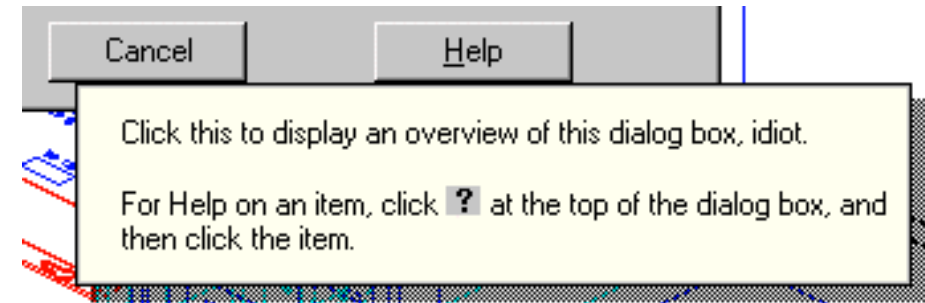
- placer le message d'erreur là où l'utilisateur est censé regarder
- mettre en évidence le champ erroné
- afficher des messages d'erreur explicites, brefs, non réprobateurs et auto-suffisants

Traitement des erreurs (suite)

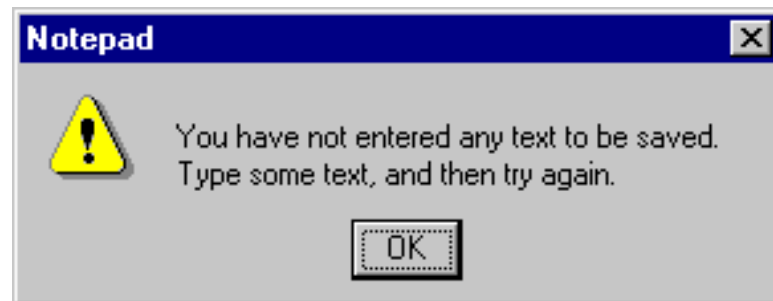
Les messages d'erreur sont importants



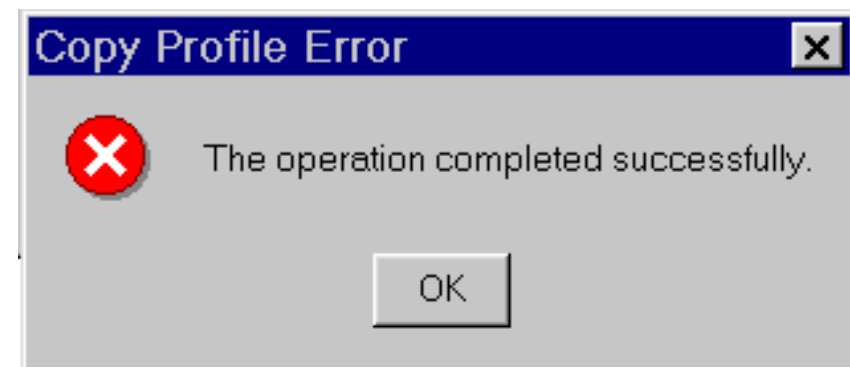
Adobe's *ImageReady*



AutoCAD *Mechanical*



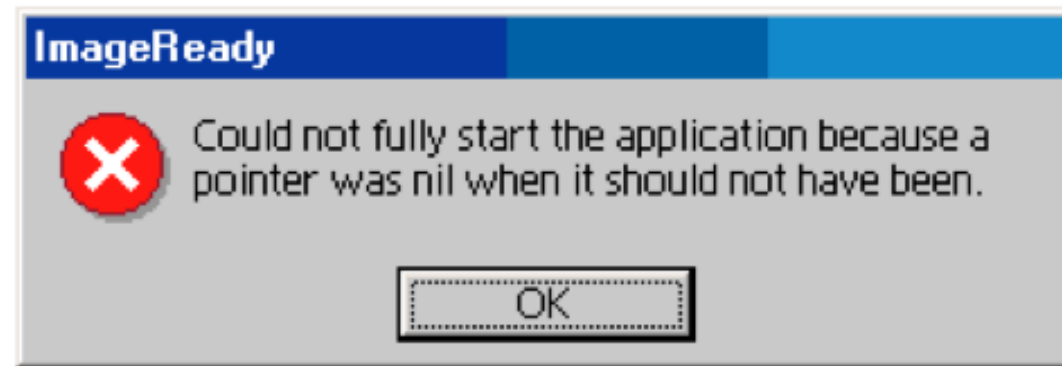
Windows *Notepad*



Microsoft's *NT Operating System*

Traitement des erreurs (suite)

Les messages d'erreur sont importants



Le message doit être rédigé dans des termes compréhensibles par l'utilisateur, si possible reliés à la tâche qu'il essayait d'accomplir

Eviter de donner l'impression à l'utilisateur qu'il est stupide

Essayer d'inclure une façon de corriger l'erreur

- Exemple : manque de place -> vider la corbeille

Traitement des erreurs (suite)

L'utilisateur n'aime pas se sentir "piégé »

- Engagement prématuré : forcer à prendre une décision trop tôt
- Essayer de toujours lui laisser une porte de sortie clairement identifiée

Solutions :

- bouton Annuler (*cancel*)
- bouton Reset/Default pour annuler des changements
- commande Défaire (*undo*), avec historique, et Refaire (*redo*)
- commande Interrompre (et/ou *Pause*)
- commande Quitter

Traitement des erreurs (suite)

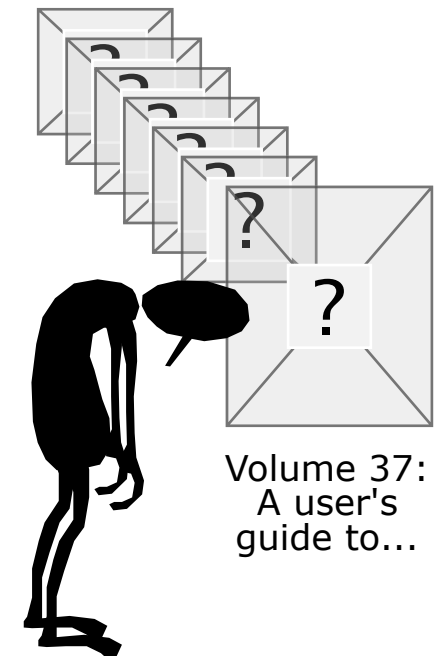
Prévoir un **système d'aide**

Remarque : le système d'aide n'est pas là pour aider à faire passer une interface mal conçue

La plupart des utilisateurs ne lisent pas la documentation

Situation la plus courante : le « mode panique »

- besoin ponctuel, spécifique et urgent
- nécessite une documentation en ligne avec des mécanismes d'indexation



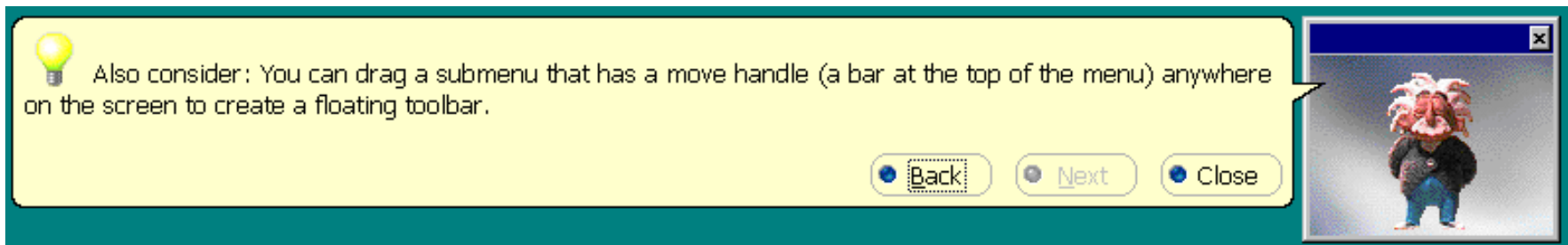
Traitement des erreurs (suite)

L'aide peut prendre différentes formes :

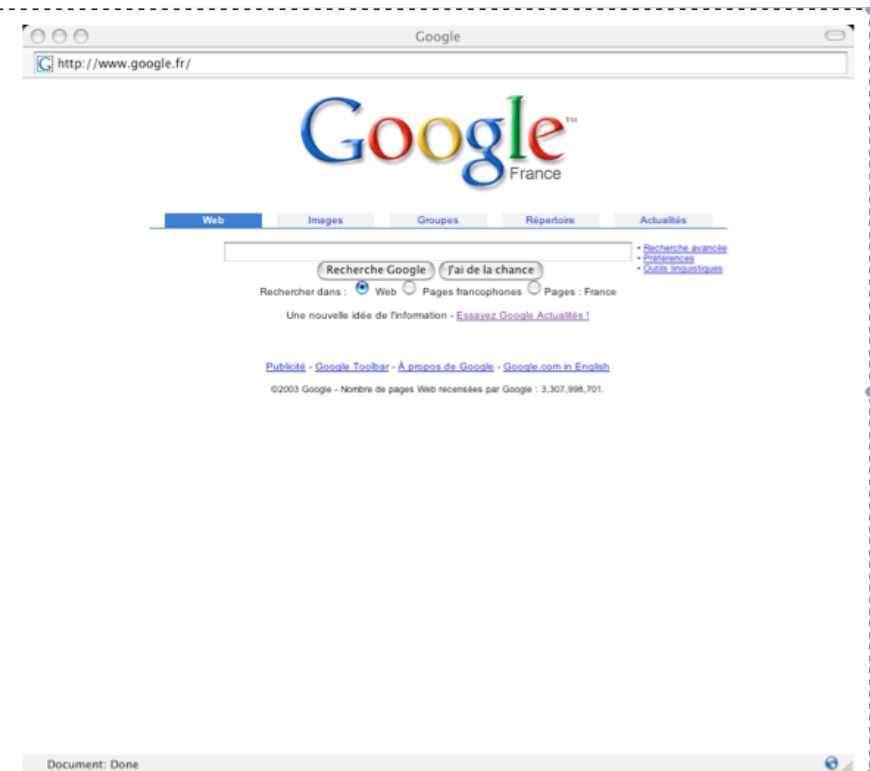
- tutoriels, démonstrations (« comment créer un tableau »)
- manuel de référence (table des matières, index, etc.)
- mémento (cartes de référence, tip of the day, tooltips, glossaire)
- aide contextuelle (qu'est ce que c'est ?)
- assistants

Exemples d'aide contextuelle :

- où suis-je ?
- où est...?
- et maintenant ?
- que s'est-il passé ?
- pourquoi est ce que ça ne...?



Concision



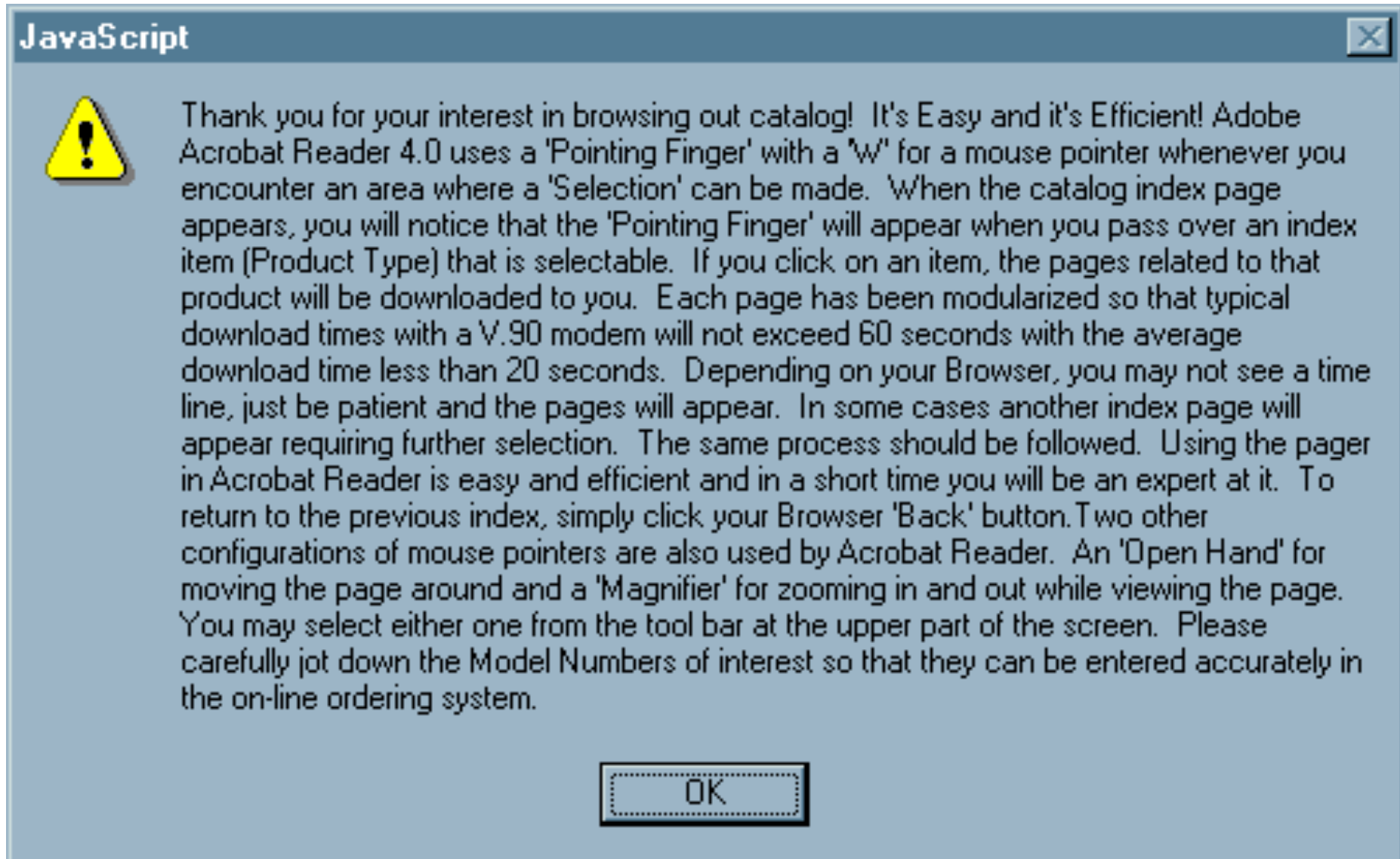
Concision

Ensemble des moyens visant à réduire la charge perceptive et mnésique de l'utilisateur

Recommandations :

- n'afficher que les informations pertinentes (mais cela dépend de ce que veut faire l'utilisateur)
- réduire le nombre d'actions élémentaires pour atteindre un objectif donné
- minimiser les saisies/lectures
- éviter les textes trop verbeux
- ne pas demander de saisir des informations qui peuvent être déduites
- éviter à l'utilisateur d'avoir à se souvenir d'informations
- ne pas l'obliger à faire des calculs qui peuvent être automatisés (parfois liée à l'attention, memorisation, externalisation de cognition)

Concision (suite)



Concision (suite)

Présenter uniquement les informations nécessaires

- « less is more » : limiter le nombre de widgets et de fenêtres qui introduisent des manipulations supplémentaires
- cacher ou supprimer les informations non essentielles
- utiliser si possible l'ordre naturel des choses

Préférer les techniques de manipulation directe aux techniques nécessitant la saisie de commande ou d'arguments
(*point and click instead of remember and type*)

Concision (suite)

Préférer un petit nombre de commandes génériques

Expliciter les contraintes existantes :

Form1

Date: []

[] [] []
Month Day Year

May 22 1997
Month Day Year

May [] 22 [] 1997 []
Month Day Year

Appointment

General Attendees Notes Planner

When

Start: 8 :30AM Wed 5 /14 /97

End: 4 :30PM Wed 5 /14 /97

☐ All day

Description:

Smart Technology Ser

Where: []

May 1997

S	M	T	W	T	F	S
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

Reconnaître est plus facile que se souvenir, mais bien souvent, en terme de représentation graphique, *less is more...*

Catégories (heuristiques)

- 1- Compatibilité
- 2- Guidage
- 3- Homogénéité
- 4- Flexibilité
- 5- Traitement des erreurs
- 6- Concision

Evaluation Heuristique - Nielsen (1990)



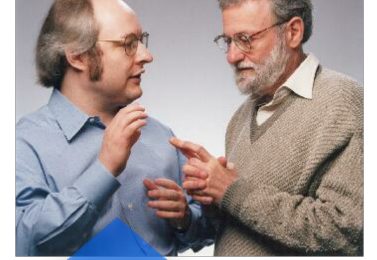
L'inspection systématique d'interface, en utilisant les catégories d'usabilité (en suite)

Procédure

- 3-5 inspecteurs (experts d'usabilité, les utilisateurs finaux)
- Inspecter l'interface (environ 1-2 heures pour les interfaces simples)
- Comparer leurs notes après

Marche pour storyboards, prototypes, et vrais systèmes

Evaluation Heuristique - Nielsen (1990)



Non guidée

- Exploration du système, sans tâche dirigée
- Aide à explorer les divers aspects de l'interface

Guidée par des Scénarios (de phase 1 ou 2)

- Utiliser des tâches représentatives des utilisateurs
- Problèmes identifiés dans les parties pertinentes
- Fonctionnalités d'intérêt sont évaluées
- Mais des problèmes peuvent échapper

Design walkthrough

But :

Aider à identifier les problèmes informellement et rapidement, en utilisant des critères d'évaluation (à vous à préciser, ex ces categories/critères de usabilité)

Procédure

- Choisir un petit groupe avec des rôles et des expertises différentes
- Fixer une durée, 1 heure au maximum
- Un présentateur déroule le scénario (storyboard, prototype vidéo)
- Choisir le niveau des critiques
- Le groupe identifie autant de problèmes que possible
- Utiliser des règles pour aider à trouver les problèmes
(ex. Principes de conception, spécifications, critères d'usabilité, ...)