

**GUIDE DE CONCEPTION D'INTERACTIONS
PERSONNES-SYSTÈMES (IPS)**

GUIDE DE STYLE

**LES ORIENTATIONS TECHNOLOGIQUES DU
RÉSEAU SOCIO SANITAIRE**

Septembre 1997



Gouvernement du Québec
Ministère de la Santé et des Services sociaux
Direction générale des technologies de l'information

Nous tenons à remercier très sincèrement toutes les personnes qui ont contribué à l'élaboration et à la production de ce document.

Dans ce document, le générique masculin est utilisé sans intention discriminatoire et uniquement dans le but d'alléger le texte.

Dépôt légal
Bibliothèque nationale du Québec
4^e trimestre 1997

ISBN : 2-550-32185-5

AVERTISSEMENT

Le présent document est le résultat d'une partie des travaux réalisés entre août 1996 et mai 1997, lors du projet d'élaboration du *Guide de conception d'interactions personnes-systèmes* du réseau sociosanitaire.

Le domaine du génie logiciel évoluant rapidement, il est possible que l'on retrouve dans ce texte quelques informations qui ne sont plus à jour. Au besoin, des précisions et des informations complémentaires seront disponibles sur le site Internet du groupe de soutien à la mise en œuvre des orientations technologiques du réseau sociosanitaire, le SMOOT, à l'adresse suivante : **<http://www.rsss.gouv.qc.ca/ot>**.

TABLE DES MATIÈRES

TERMINOLOGIE UTILISÉE.....	VII
DOCUMENTS PERTINENTS	XV
INTRODUCTION	1
1. L'ENVIRONNEMENT WINDOWS	3
1.1 LE BUREAU	3
1.2 FENÊTRES.....	5
2. DISPOSITIFS D'INTERACTION	7
2.1 LA SOURIS	7
2.1.1 <i>Pointeurs de la souris</i>	7
2.1.2 <i>Actions de la souris</i>	9
2.2 LE CLAVIER.....	9
2.2.1 <i>Touches de texte</i>	10
2.2.2 <i>Touche d'accès rapide</i>	10
2.2.3 <i>Touches de mode</i>	11
2.2.4 <i>Raccourci au clavier</i>	11
2.3 AUTRES DISPOSITIFS D'INTERACTION	12
2.3.1 <i>Reconnaissance de la parole</i>	12
2.3.2 <i>Écran tactile</i>	12
2.3.3 <i>Reconnaissance optique des caractères</i>	12
2.3.4 <i>Ardoise électronique</i>	13
2.3.4.1 <i>L'entrée de données avec le stylet</i>	13
2.3.4.2 <i>Les pointeurs</i>	14
2.3.4.3 <i>La sélection avec le stylet</i>	15
2.3.4.4 <i>L'utilisation des gestes avec le stylet (<i>pen gestures</i>)</i>	16
2.3.4.5 <i>La reconnaissance de l'écriture</i>	17
2.3.4.6 <i>Le stockage d'encre</i>	18
2.3.4.7 <i>Positionnement du curseur</i>	18
2.3.4.8 <i>Techniques d'édition spécifiques du stylet</i>	19
3. TECHNIQUES D'INTERACTION	23
3.1 TRANSFORMATIONS.....	23
3.2 OPÉRATIONS DE TRANSFERT.....	25
3.2.1 <i>Méthode des commandes</i>	25
3.2.2 <i>Méthode de manipulation directe</i>	27

3.2.3	<i>Retour d'information de transfert.....</i>	31
3.2.4	<i>Commandes de transfert spécialisées.....</i>	33
3.2.5	<i>Raccourci au clavier pour les opérations de transfert.....</i>	34
3.3	<i>Opérations de création</i>	34
3.3.1	<i>Commande Copier</i>	34
3.3.2	<i>Commande Nouveau</i>	35
3.3.3	<i>Commande Insérer.....</i>	35
3.3.4	<i>Utilisation des contrôles.....</i>	35
4.	FENÊTRES.....	37
4.1	TYPES DE FENÊTRES	37
4.2	COMPOSANTES DE LA FENÊTRE PRINCIPALE	37
4.3	OPÉRATIONS WINDOWS DE BASE.....	38
4.3.1	<i>Activation et désactivation.....</i>	39
4.3.2	<i>Ouverture et fermeture des fenêtres</i>	39
4.3.3	<i>Déplacement des fenêtres</i>	39
4.3.4	<i>Redimensionnement des fenêtres</i>	39
4.3.5	<i>Défilement des fenêtres.....</i>	40
4.3.6	<i>Fractionnement des fenêtres.....</i>	40
5.	FENÊTRES SECONDAIRES	41
5.1	CARACTÉRISTIQUES DES FENÊTRES SECONDAIRES	41
5.1.1	<i>Apparence et comportement</i>	41
5.1.2	<i>Positionnement des fenêtres</i>	43
5.1.3	<i>Fenêtre modale vs non modale</i>	43
5.1.4	<i>Boutons par défaut.....</i>	44
5.1.5	<i>Navigation dans les fenêtres secondaires.....</i>	44
5.1.6	<i>Validation des données</i>	45
5.2	BOÎTES DE DIALOGUE	46
5.2.1	<i>Commandes des boîtes de dialogue</i>	46
5.2.2	<i>Disposition.....</i>	46
5.2.3	<i>Interfaces de boîtes de dialogue répandues</i>	47
5.3	BOÎTES DE MESSAGE.....	53
5.3.1	<i>Étiquette de la barre de titre.....</i>	53
5.3.2	<i>Types de boîtes de message</i>	53

5.3.3	Textes de boîte de message	54
6.	GESTION DE L'ENVIRONNEMENT FENÊTRÉ	57
6.1	INTERFACE À DOCUMENT UNIQUE	57
6.2	INTERFACE À DOCUMENTS MULTIPLES	57
7.	MENUS, COMMANDES ET BARRES D'OUTILS.....	61
7.1	MENUS.....	61
7.1.1	Barre de menu et menus déroulants	61
7.1.2	Menus déroulants communs	63
7.1.3	Menus en cascade	66
7.1.4	Titres de menu.....	66
7.1.5	Éléments de menu	67
7.2	CONTRÔLES	70
7.2.1	Boutons	70
7.2.2	Zones de liste	72
7.2.3	Champs de texte.....	76
7.2.4	Onglets.....	79
7.3	BARRES D'OUTILS ET BARRES D'ÉTAT	80
8.	CONCEPTION VISUELLE DES OBJETS D'INTERFACE	81
8.1	POLICES ET TAILLES DES CARACTÈRES	81
8.2	MAJUSCULES.....	82
8.3	REGROUPEMENT ET ESPACEMENT	82
8.4	ALIGNEMENT.....	83
8.5	DISPOSITION	83
9.	CONCEPTION D'IMAGES GRAPHIQUES.....	85
9.1	CONCEPTION DES ICÔNES.....	85
9.2	CONCEPTION DES POINTEURS	86
9.3	APPARENCE SÉLECTIONNÉE DES OBJETS	88
10.	AIDE À L'UTILISATEUR	89
10.1	ACCÈS À L'AIDE.....	89
10.2	MENU ? (AIDE)	89
10.2.1	Contenu standard du menu ? (Aide).....	90
10.2.2	Contenu facultatif du menu ? (Aide).....	91

10.3 TOUCHE D'AIDE	91
10.4 MODE D'AIDE.....	91
10.5 AIDE CONTEXTUELLE : BARRES D'ÉTAT.....	91
10.5.1 Messages de barres d'état.....	91
10.5.2 Recommandations pour la rédaction de messages de barres d'état.....	92
10.6 AIDE DANS LES BOÎTES DE DIALOGUE ET DE MESSAGE.....	92
CONCLUSION.....	95
ANNEXE 1 – COLLABORATIONS.....	99

LISTE DES FIGURES ET DES TABLEAUX

Liste des figures

Figure 1 : Le bureau	4
Figure 2 : Différents usages des fenêtres.....	5
Figure 3 : Geste <i>Lasso-Taper</i>	15
Figure 4 : Les gestes d'édition avec le stylet.....	17
Figure 5 : Positionnement de l'entrée d'écriture manuscrite	18
Figure 6 : Zone de texte standard avec bouton d'outil d'écriture	19
Figure 7 : Ajout du bouton d'outil d'écriture.....	19
Figure 8 : Boîte de texte affichant un opérateur de sélection.....	19
Figure 9 : Fenêtres d'outil d'écriture uniligne et multiligne	20
Figure 10 : Sélection de texte avec l'opérateur de sélection.....	20
Figure 11 : Opérateur d'action avec liste d'options	20
Figure 12 : Affichage d'une liste d'options par tape	21
Figure 13 : Accès au menu autonome par tape.....	21
Figure 14 : Fenêtre autonome de l'outil clavier à l'écran	22
Figure 15 : Autre forme de l'opération <i>Glisser-Déposer</i>	28
Figure 16 : Défilement automatique en fonction de la vitesse de déplacement du pointeur.....	31
Figure 17 : Représentations de contour et translucides pour les opérations de transfert.....	32
Figure 18 : Pointeur <i>Ne peut déplacer l'objet à cet emplacement</i>	33
Figure 19 : Fenêtre principale	38
Figure 20 : Fenêtre secondaire	41
Figure 21 : Boîte de dialogue <i>Ouvrir</i>	48
Figure 22 : Boîte de dialogue <i>Enregistrer sous</i>	49
Figure 23 : Boîtes de dialogue <i>Rechercher et Remplacer</i>	50
Figure 24 : Boîte de dialogue <i>Imprimer</i>	50
Figure 25 : Boîte de dialogue <i>Configuration de l'impression</i>	51
Figure 26 : Boîte de dialogue <i>Mise en page</i>	52
Figure 27 : Boîte de dialogue <i>Caractères</i>	52
Figure 28 : Boîte de message.....	53
Figure 29 : Fenêtre parent et enfant MDI.....	58
Figure 30 : Barre de menu	61
Figure 31 : Exemple d'un menu <i>Fichier</i>	63
Figure 32 : Exemple d'un menu <i>Edition</i>	64
Figure 33 : Exemple d'un menu <i>Affichage</i>	65
Figure 34 : Exemple d'un menu <i>Fenêtre</i>	65
Figure 35 : Exemple d'un menu ?	65
Figure 36 : Touches d'accès rapide dans une barre de menu	67
Figure 37 : Formats pour différents éléments de menu.....	67
Figure 38 : Séparateur inadéquat.....	68
Figure 39 : Boutons de commande.....	71
Figure 40 : Exemples d'apparence enfoncée et normale d'un bouton de commande.....	71
Figure 41 : Ensemble de cases à option.....	71
Figure 42 : Ensemble de cases à cocher	72
Figure 43 : Liste à sélection unique.....	73
Figure 44 : Liste déroulante (états fermé et ouvert).....	74

Figure 45 : Liste à sélection multiple	75
Figure 46 : Liste modifiable	75
Figure 47 : Liste déroulante modifiable (états fermé et ouvert).....	76
Figure 48 : Arborescence.....	76
Figure 49 : Zone de texte standard	77
Figure 50 : Zone de texte RTF	78
Figure 51 : Zone de texte statique	78
Figure 52 : Compteur.....	79
Figure 53 : Onglets	79
Figure 54 : Exemple de barres d'outils	80
Figure 55 : Exemple de barre d'état	80
Figure 56 : Disposition recommandée et espacement des contrôles et du texte	82
Figure 57 : Exemple de disposition de boutons de commande	83
Figure 58 : Icônes en trois tailles.....	85
Figure 59 : Points d'impact des pointeurs.....	87
Figure 60 : Exemple d'un objet d'apparence surlignée	88
Figure 61 : Menu ? (<i>Aide</i>)	90
Figure 62 : Message de barre d'état décrivant un élément de menu.....	92
Figure 63 : Bouton de commande <i>Aide</i> dans une fenêtre secondaire.....	93

Liste des tableaux

Tableau 1 : Pointeurs courants	8
Tableau 2 : Mouvements fondamentaux du stylet.....	13
Tableau 3 : Types de messages et symboles associés	54
Tableau 4 : Éléments standard du menu ? (<i>Aide</i>).....	90

TERMINOLOGIE UTILISÉE

Aide contextuelle :

Information sur un objet et sur son état actuel. Répond aux questions « Qu'est-ce que c'est ? » et « À quoi ça sert ? ».

Barre de menu :

Barre horizontale, située au sommet d'une fenêtre, sous la barre de titre, qui contient des menus.

Barre d'état :

Espace qui permet l'affichage d'informations sur l'état du contenu d'une fenêtre. Généralement placée au bas de la fenêtre en question (*status bar*).

Barre d'outils :

Cadre ou espace spécial qui contient un ensemble de contrôles (*toolbar*).

Boîte de dialogue :

Fenêtre secondaire qui recueille de l'information additionnelle de l'utilisateur (*dialog box*).

Boîte de message :

Fenêtre secondaire affichée pour informer l'utilisateur de conditions particulières (*message box*).

Boucler :

Fonction permettant de reporter le texte en fin de ligne au début de la ligne suivante (*wrap*). Elle permet de lire un texte de manière continue, sans que des sections de phrases ne soient cachées par les bordures d'une boîte de texte, par exemple.

Bouton de commande :

Contrôle Windows standard qui amorce une commande ou fixe une option.

Bouton par défaut :

Bouton de commande activé lorsque l'utilisateur appuie sur la touche ENTRÉE. Un bouton par défaut apparaît généralement dans les fenêtres secondaires (*default button*).

Bureau :

Aire de travail visuelle qui occupe tout l'espace d'affichage. Le bureau est également un contenant et peut être utilisé pour placer des objets (*desktop*).

Case à cocher :

Contrôle standard Windows qui affiche une option, cochée (sélectionnée) ou non (*check box*).

Case à option :

Contrôle standard Windows qui permet à l'utilisateur de sélectionner parmi un ensemble d'options mutuellement exclusives (*option button* ou *radio button*).

Champ :

Terme utilisé pour désigner l'ensemble des objets comprenant une zone où l'utilisateur peut saisir de l'information sous forme de texte (zone de liste modifiable, zone de liste déroulante, zone de texte).

Compteur :

Contrôle composé d'une zone de texte et de boutons d'incrément et de décrément qui permettent à l'utilisateur de choisir une valeur dans un intervalle fini de valeurs (*spin box*).

Contenant :

Objet qui contient d'autres objets.

Contrôle :

Objet qui permet l'interaction avec l'utilisateur ou l'entrée de données. Sert souvent à amorcer une action, afficher de l'information ou fixer des valeurs.

Coquille :

Terme générique qui fait référence à l'interface permettant à l'utilisateur de contrôler le système (*shell*).

Curseur :

Symbole indiquant l'endroit où l'on peut entrer des données.

Curseur de fractionnement :

Contrôle spécial ajouté à une fenêtre, généralement adjacent aux barres de défilement, qui permet à l'utilisateur de fractionner une fenêtre en panneaux ou encore d'ajuster le fractionnement (*split box*).

Défilement :

Action de déplacer la portion visible d'un objet pour en visualiser une autre partie.

Dialogue :

Suite structurée d'interactions entre l'utilisateur et le système. L'affichage est un élément de dialogue dans le sens système-utilisateur ; la saisie est un élément dans le sens utilisateur-système.

Dispositif d'interaction :

Dispositif physique à l'aide duquel l'utilisateur saisit ou fait afficher de l'information. Le clavier et la souris sont les exemples les plus connus.

Document :

Unité de données (généralement un fichier) dont l'utilisateur se sert pour ses tâches. Peut être échangé entre les utilisateurs.

Esquisser :

Faire un « geste » avec le stylet afin d'exécuter une opération.

Étiquette :

Texte destiné à déterminer par sa dénomination la nature de l'information saisie ou affichée dans l'objet auquel l'étiquette est associée (ex. : l'étiquette en avant ou au-dessus d'une zone de texte indique la nature de l'information à saisir ou affichée dans cette zone), (*label*).

Fenêtre :

Objet standard Windows qui affiche de l'information. Une fenêtre est un espace de l'écran contrôlable séparément et possédant généralement une bordure rectangulaire.

Fenêtre active :

Fenêtre dans laquelle l'utilisateur travaille ou entre des données. Généralement située par-dessus les autres fenêtres, selon l'ordre en Z. Se distingue par sa barre de titre d'une couleur différente de celle des fenêtres inactives.

Fenêtre autonome :

Fenêtre secondaire, sans barre de titre, affichée à côté d'un objet : elle présente de l'information contextuelle sur l'objet auquel elle est associée (*pop-up window*).

Fenêtre modale :

Interaction restreinte par un mode donné. Se dit d'une fenêtre secondaire qui restreint l'interaction de l'utilisateur avec d'autres fenêtres. Une fenêtre peut être modale par rapport à sa fenêtre principale ou par rapport à tout le système.

Fenêtre non modale :

Interaction non restrictive. Se dit d'une fenêtre secondaire qui ne limite pas l'interaction de l'utilisateur avec les autres fenêtres ou avec le système.

Fenêtre principale :

Fenêtre dans laquelle l'interaction principale a lieu.

Fenêtre secondaire :

Fenêtre qui offre de l'information ou de l'interaction supplémentaire associée à des objets de la fenêtre principale.

Geste :

Dessin à main levée tracé à l'aide du stylet, qui laisse une trace sur l'écran sous forme d'encre électronique (*pen gesture*). Sert à exécuter certaines commandes telles que *Copier* ou *Effacer*.

Glisser :

Action qui consiste à déplacer un objet d'un emplacement à l'autre dans une interface. Pour faire glisser un objet, il faut d'abord appuyer sur un bouton de la souris et le maintenir enfoncé, tout en déplaçant l'objet (*drag*).

Glisser-Déposer :

Technique pour déplacer, copier ou lier un objet en le glissant. La destination permet d'interpréter l'opération (*drag-and-drop*).

Interface à documents multiples :

Technique permettant de gérer un ensemble de fenêtres où les documents sont ouverts dans des fenêtres (appelées *fenêtres enfants*) qui sont subordonnées à une seule fenêtre principale (aussi appelée *fenêtre parent*), (*multiple document interface – MDI*).

Interface à document unique :

Technique selon laquelle l'interface ne représente et ne traite qu'une seule instance à la fois (*Single document interface – SDI*).

Manipulation directe :

Technique d'interaction à l'aide de laquelle l'utilisateur applique des actions directement sur les objets visibles de l'IPS. La technique *Glisser-Déposer* est un exemple de manipulation directe.

MDI :

Voir *Interface à documents multiples*.

Menu :

Liste textuelle ou graphique d'options parmi lesquelles un utilisateur peut choisir.

Menu autonome :

Menu affiché à l'emplacement d'un objet qui contient des commandes pertinentes dans le contexte de la sélection. Aussi appelé menu contextuel (*pop-up menu*).

Menu déroulant :

Menu affiché à partir de la barre de menu (*drop-down menu*).

Menu en cascade :

Sous-menu d'un élément de menu. Aussi appelé *menu hiérarchique*, *menu enfant* ou *sous-menu*.

Mode :

État d'interaction particulier, souvent exclusif à d'autres formes d'interaction.

Objet :

Entité ou composante, identifiable par un utilisateur, qui se distingue par ses propriétés, les opérations qui lui sont associées ainsi que les liens qu'elle entretient avec les autres objets.

Objet OLE incorporé :

Objet donné qui garde les fonctionnalités d'édition et d'opération originales de l'application l'ayant créé, alors qu'il réside physiquement dans un autre document.

Ordre en Z :

Disposition d'objets à l'écran, telles des fenêtres, en couches superposées.

Palette :

Fenêtre secondaire non modale affichant une barre d'outils ou d'autres choix, tels que des couleurs ou des textures.

Panneau :

Un des espaces dans une fenêtre fractionnée à l'aide d'un curseur de fractionnement.

Poignée :

Élément d'interface ajouté à un objet qui offre un point de contrôle pour déplacer, redimensionner, refaçonner ou effectuer toute autre opération se rapportant à cet objet (*handle*).

Point d'impact :

Portion précise du pointeur qui définit l'emplacement exact ou l'objet sur lequel pointe l'utilisateur (*hot spot*).

Pointeur :

Image graphique qui sert à pointer sur un objet de l'interface à l'aide d'un dispositif d'entrée (*Pointer*). Parfois appelé *curseur*.

Presse-Papiers :

Espace de rangement pour les objets, les données ou leurs références, après qu'un utilisateur a effectué une commande *Couper* ou *Copier* (*clipboard*).

Raccourci au clavier :

Action au clavier (touche de fonction, touche du clavier ou combinaison de touches) qui active une commande donnée (*shortcut keys*).

Saisie :

Fait référence aux actions de l'utilisateur, soit pour déclencher une action (sélection d'un élément dans un menu, d'un bouton de commande, etc.), soit pour entrer et éditer du texte (saisie textuelle).

SDI :

Voir *Interface à document unique*.

Technique d'interaction :

Moyen par lequel l'utilisateur interagit avec le système (sélection de menu, langage de commande, etc.).

Transformation :

Unité de changement apportée à un objet, telle que modifier ses attributs ou encore le copier.

Utilisabilité :

Caractère utilisable d'une IPS, qui se mesure par des critères comme la facilité d'apprentissage, la vitesse d'utilisation, la capacité de l'utilisateur de travailler sans faire d'erreurs et la satisfaction d'utilisation. L'utilisabilité résulte de l'adaptation de l'interaction aux besoins des utilisateurs et à la tâche. Ce nouveau terme remplace l'expression *convivialité*.

Zone de liste :

Contrôle standard Windows affichant une liste d'options (*list box*).

Zone de liste à sélection unique :

Zone de liste qui ne permet la sélection que d'un seul élément de la liste (*single selection list box*).

Zone de liste modifiable :

Contrôle standard Windows combinant une zone de texte et une zone de liste interdépendantes (*combo box*).

Zone de liste déroulante fixe :

Contrôle standard Windows affichant une valeur ou une option donnée, mais qui peut être ouvert pour afficher un ensemble d'options (*drop-down list*).

Zone de liste déroulante modifiable :

Contrôle standard Windows combinant une zone de texte et une zone de liste déroulante fixe (*drop-down combo box*).

Zone de texte :

Contrôle standard Windows dans lequel un utilisateur peut entrer et modifier du texte (*text box*).

Zone sensible :

Aire d'interaction d'un objet avec lequel le point d'impact du pointeur doit entrer en contact (*hot zone*).

DOCUMENTS PERTINENTS

[GER97] :

MINISTÈRE DE LA SANTÉ ET DES SERVICES SOCIAUX DU QUÉBEC, Direction générale des technologies de l'information. *Guide de conception d'interactions personnes-systèmes : Guide d'ergonomie*, 1997.

[MET97] :

MINISTÈRE DE LA SANTÉ ET DES SERVICES SOCIAUX DU QUÉBEC, Direction générale des technologies de l'information. *Guide de conception d'interactions personnes-systèmes : Méthode de conception ergonomique d'IPS*, 1997.

[WIG92] :

MICROSOFT. *The Windows Interface : An Application Design Guide*, Microsoft Press, 1992.

[WIG95] :

MICROSOFT. *The Windows Interface Guidelines for Software Design*, Microsoft Press, 1995.

INTRODUCTION

Ce guide de style est produit dans le cadre du projet *Guide de conception d'interactions personnes-systèmes* (IPS) du réseau de la santé et des services sociaux (RSSS). Il décrit les principales composantes de l'environnement Windows 3.1, ainsi que la gestion des interactions des utilisateurs avec ces composantes. En tant qu'outil de conception, il permet d'assurer la normalisation des environnements informatiques ainsi que la cohérence des choix de conception. Un guide d'ergonomie [GER97] a également été produit dans le cadre de ce projet, servant un objectif autre que celui du *Guide de style*.

Dans ce document, l'interface personne-système correspond aux éléments que l'utilisateur peut toucher, voir, entendre, etc., lorsqu'il utilise le logiciel. Un exemple d'interface personne-système est un ensemble de fenêtres où l'utilisateur peut choisir ou éditer de l'information en utilisant une souris et un clavier. Le terme *interaction* a une portée plus large et inclut les éléments de gestion du dialogue, les mécanismes de gestion des erreurs, les fonctionnalités requises pour que l'utilisateur puisse accomplir sa tâche, etc. L'interface est un outil dans l'interaction. La méthode de conception des IPS, décrite dans le document *Méthode de conception ergonomique d'interactions personnes-systèmes* [MET97], permet de concevoir et de déterminer l'interaction, même si c'est l'interface qui en est la partie la plus visible.

L'objectif d'un guide de style est avant tout de spécifier le comportement des composantes d'interface dans le contexte d'un environnement particulier (Windows, Motif, Macintosh, etc.) de façon à assurer la cohérence du style pour les logiciels développés sous cet environnement. Chaque environnement distinct possède un guide de style qui lui est propre.

L'objectif d'un guide d'ergonomie est avant tout d'aider à concevoir des interactions appropriées au regard des tâches qu'elles supportent, plus conviviales et moins enclines à l'erreur. Les principes et recommandations ergonomiques qui y sont énoncés et qui guident la conception sont indépendants de l'environnement choisi.

Le *Guide de style* s'adresse aux concepteurs, aux développeurs et aux autres personnes engagées dans la conception d'IPS pour le RSSS, donc à un public déjà familiarisé avec la terminologie utilisée dans ce domaine.

En décrivant les composantes de l'interface et leur comportement, le *Guide de style* favorise la réutilisation de concepts fonctionnels normalisés, tout en permettant l'adaptation de ces concepts aux exigences liées aux différents types de tâches à effectuer à l'intérieur du RSSS. Le *Guide de style* a un objectif essentiellement descriptif et n'incorpore pas de recommandations ergonomiques de conception d'interactions : c'est pourquoi il est utilisé de concert avec le *Guide d'ergonomie*, qui détermine les principes et les recommandations ergonomiques améliorant la performance et la convivialité des interactions. Les tables des matières des deux documents ont d'ailleurs été harmonisées dans le but d'en faciliter la consultation.

Ce guide de style a été conçu à partir des guides de style de Microsoft pour l'environnement Windows [WIG95], [WIG92]. Il en reprend les principaux éléments en fonction des besoins spécifiques des appli-

cations informatiques propres au RSSS. Même si le document correspond en majeure partie aux éléments de style de Windows 3.1, il incorpore des éléments de style de Windows 95 et de Windows NT 4.0, de façon à ce que la transition vers ces versions plus récentes de Windows se fasse harmonieusement. Si cette transition devait se faire, il suffirait d'incorporer à ce guide les éléments de style propres à Windows 95 ou à Windows NT 4.0. Les éléments présents dans ce guide de style permettent à une interface conçue pour être exploitée sous Windows 3.1 d'être adaptée avec peu de changements à Windows 95 ou à Windows NT 4.0.

Ce document présente des informations sur les sujets suivants :

- l'environnement Windows;
- les dispositifs d'interaction;
- les techniques d'interaction;
- les fenêtres principales;
- les fenêtres secondaires;
- la gestion de l'environnement fenêtré;
- les menus, les commandes et les barres d'outils;
- la conception visuelle des objets d'interface;
- la conception des images graphiques;
- l'aide à l'utilisateur.

1. L'ENVIRONNEMENT WINDOWS

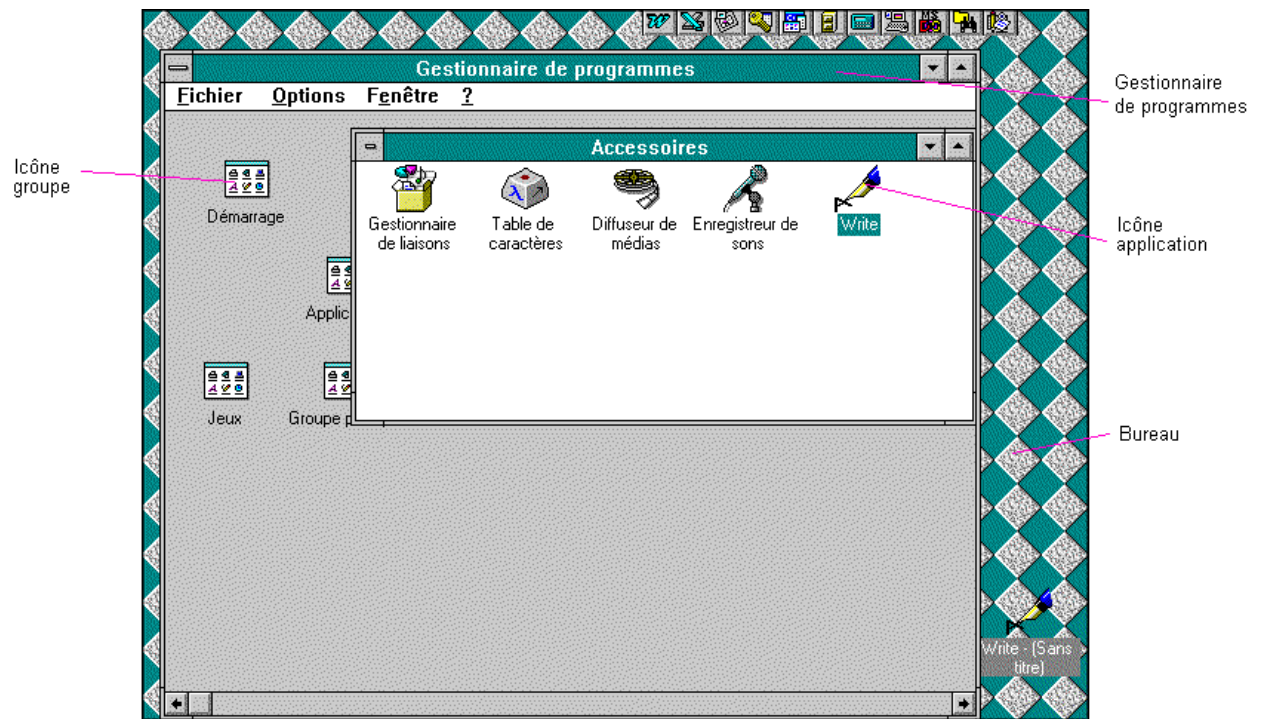
Cette section présente un bref survol de certains des éléments de base inclus dans le système d'exploitation Microsoft Windows. Ces éléments permettent à l'utilisateur de contrôler l'environnement (auquel on fait référence parfois sous le nom de *coquille*). Ils fournissent non seulement la toile de fond de l'environnement de travail mais peuvent également servir de points de repère pour l'interaction des utilisateurs avec votre application.

1.1 LE BUREAU

Le bureau représente l'espace de travail principal de l'utilisateur : il remplit l'écran et forme l'arrière-plan visuel sur lequel l'utilisateur peut ouvrir des fenêtres. Dans l'environnement Windows 3.1, l'application principale est le *Gestionnaire de programmes*, fenêtre principale qui regroupe toutes les applications d'un même système. Le *Gestionnaire de programmes* peut contenir d'autres fenêtres principales, soit les fenêtres appelées *Groupes de programmes*, qui contiennent un ensemble de programmes, soit les fenêtres principales ou secondaires des applications.

Lorsqu'elles ne sont pas ouvertes, les fenêtres *Groupe de programmes* sont représentées sous la forme d'une icône standard. Lorsque les applications ne sont pas ouvertes, elles sont également représentées sous la forme d'une icône qui leur est propre, à l'intérieur des fenêtres *Groupes de programmes*. Les fenêtres des applications peuvent également être ouvertes directement à l'arrière-plan, plutôt qu'à l'intérieur du *Gestionnaire de programmes*.

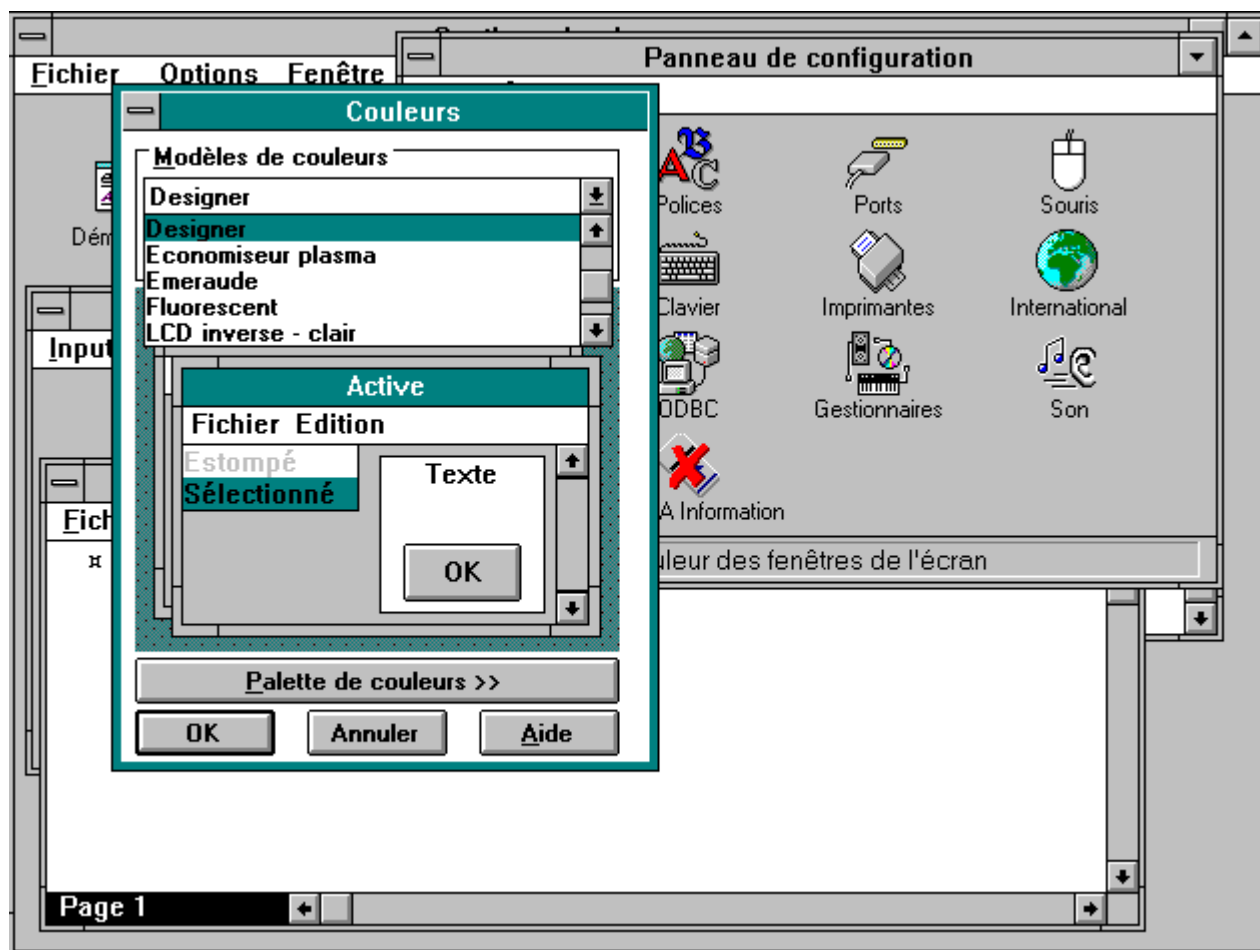
Figure 1 : Le bureau



1.2 FENÊTRES

Les fenêtres de l'environnement Windows permettent de visualiser et d'éditer de l'information, ainsi que de visualiser le contenu et les propriétés des objets qu'elles contiennent. Elles sont également utilisées dans l'affichage des paramètres employés pour compléter les commandes, les palettes de contrôle ou les messages d'information destinés à l'utilisateur. La figure suivante présente différents usages des fenêtres.

Figure 2 : Différents usages des fenêtres



2. DISPOSITIFS D'INTERACTION

Un utilisateur peut interagir avec les objets de l'interface à l'aide de différents dispositifs d'interaction. Les deux dispositifs les plus répandus sont la souris et le clavier. Cette section couvre l'essentiel du comportement de base de ces deux dispositifs.

2.1 LA SOURIS

Dans Microsoft Windows, la souris est le dispositif privilégié pour l'interaction avec les objets de l'interface.

2.1.1 Pointeurs de la souris

La souris est directement liée à une icône sur l'écran appelé le *pointeur*. En positionnant le pointeur et en cliquant sur les boutons de la souris, l'utilisateur peut sélectionner des objets et les opérations qui leur sont associées.

L'apparence du pointeur peut changer en fonction de sa position sur l'écran, afin de retourner de l'information sur un emplacement dans l'interface, une opération ou un état en particulier. Le tableau suivant présente quelques formes de pointeurs courants et leur usage.

Tableau 1 : Pointeurs courants

Forme	Emplacement sur l'écran	Action disponible ou courante
	Sur la plupart des objets	Pointer, sélectionner ou déplacer
	Sur du texte	Sélectionner du texte
	Sur n'importe quel objet	Opération en cours
	Sur la plupart des objets	Mode d'aide contextuelle
	Dans une fenêtre	Agrandir l'affichage
	Sur une bordure de dimensionnement	Redimensionner la bordure verticalement
	Sur une bordure de dimensionnement	Redimensionner la bordure horizontalement
	Sur une bordure de dimensionnement	Redimensionner la bordure diagonalement
	Sur une bordure de dimensionnement	Redimensionner la bordure diagonalement
	Sur des séparateurs de colonnes	Redimensionner une colonne
	Sur des séparateurs de rangées	Redimensionner une rangée
	Sur le bouton de fractionnement	Fractionner une fenêtre horizontalement (ou une barre de défilement pour ajuster les proportions)
	Sur le bouton de fractionnement	Fractionner une fenêtre verticalement (ou une barre de défilement pour ajuster les proportions)
	Sur n'importe quel objet	Ne peut déplacer l'objet à cet emplacement

Chaque pointeur possède une zone particulière, appelée *point d'impact*, qui définit l'emplacement exact de la souris. Le point d'impact détermine l'objet influencé par les actions de la souris. Les objets à l'écran peuvent également posséder une « zone sensible », qui définit l'espace dans lequel le point d'impact doit se maintenir pour être situé sur l'objet. Généralement, la zone sensible coïncide avec la bordure d'un objet ; toutefois, elle peut également être plus grande ou plus petite, afin de faciliter l'interaction avec l'utilisateur.

2.1.2 Actions de la souris

Les actions de base de la souris sont effectuées à l'aide du bouton 1 ou du bouton 2. Par défaut, le bouton 1 est le bouton situé à l'extrême gauche de la souris et le bouton 2 est situé à l'extrême droite. Le système Windows permet d'interchanger l'appariement des boutons, en fonction des préférences de l'utilisateur (ex. : pour accommoder les gauchers). Généralement, le bouton 2 permet de reproduire plus efficacement des actions déjà disponibles avec le bouton 1. Par exemple, le bouton 2 de la souris peut être programmé pour exécuter l'équivalent d'un double-clic du bouton 1 avec un seul clic du bouton 2.

Note : Pour les souris comportant trois boutons, le bouton 2 est celui situé à l'extrême droite et non pas celui situé au centre.

Les actions suivantes sont couramment exécutées avec la souris :

Action	Description
Pointer	Positionner le pointeur de manière à ce qu'il s'arrête sur un objet précis de l'écran, sans enfoncer le bouton de la souris. <i>Pointer</i> est normalement préalable à un autre type d'interaction, tel que <i>Cliquer</i> . Cette action permet de fournir à l'utilisateur un signal visuel ou un autre type de retour d'information.
Cliquer	Pointer sur un objet, appuyer sur le bouton de la souris puis le relâcher. Généralement, il ne faut pas déplacer la souris lorsque l'on clique; on relâche ensuite rapidement le bouton après l'avoir enfoncé. <i>Cliquer</i> permet de sélectionner ou d'activer un objet.
Double-Cliquer	Positionner le pointeur sur un objet, appuyer sur le bouton puis le relâcher rapidement deux fois de suite. <i>Double-Cliquer</i> sur un objet permet généralement d'exécuter son opération par défaut.
Appuyer	Positionner le pointeur sur un objet, appuyer sur le bouton de la souris et le maintenir enfoncé.
Déplacer	Positionner le pointeur sur un objet, appuyer sur le bouton de la souris et la déplacer en maintenant le bouton enfoncé : l'objet sélectionné se déplace avec le pointeur. <i>Déplacer</i> est utilisé pour la sélection et la manipulation directe des objets.

Pour la plupart des interactions avec la souris, appuyer sur le bouton permet seulement d'identifier une opération. À ce moment, un retour d'information est généralement fourni à l'utilisateur. C'est le relâchement du bouton qui active (exécute) l'opération. Une fonction auto répétitive, telle que le fait d'appuyer sur une flèche de la barre de défilement pour faire défiler l'affichage de manière continue, est une exception.

2.2 LE CLAVIER

Le clavier est le principal moyen pour entrer ou éditer de l'information sous forme de texte. Toutefois, l'interface Windows permet également l'utilisation du clavier pour naviguer, basculer entre différents modes, modifier des données et, en guise de raccourci, accéder à certaines opérations.

Les actions suivantes sont couramment exécutées avec le clavier :

Action	Description
Appuyer	Appuyer puis relâcher une touche du clavier. Contrairement à la souris, une opération est activée par une touche du clavier après que cette dernière a été enfoncée. <i>Appuyer</i> décrit généralement l'interaction avec le clavier qui permet d'activer des commandes précises ou de naviguer.
Maintenir	
Appuyer sur et maintenir une touche.	<i>Maintenir</i> décrit généralement l'interaction avec des touches telles que ALT, MAJ (SHIFT) et CTRL, touches qui modifient le comportement normal d'autres entrées, telles que d'autres touches ou encore l'action de la souris.
Taper	Taper une information textuelle au clavier.

2.2.1 Touches de texte

Les touches de textes sont les suivantes :

- touches alphanumériques (a–z, A–Z, 0–9)
- touches de ponctuation et de symboles
- touches TAB et ENTRÉE
- la barre d'espace

Pour entrer des données textuelles, il suffit d'appuyer sur une touche du clavier pour faire afficher le caractère alphanumérique correspondant à l'écran. Certaines touches, telles que TAB et ENTRÉE, ne produisent pas de résultats visibles au clavier, à l'exception de certaines vues particulières. Dans certains contextes, les touches de texte peuvent également être utilisées à des fins de navigation ou pour accéder à certaines opérations particulières.

2.2.2 Touche d'accès rapide

Une touche d'accès rapide est une touche alphanumérique (parfois appelée mnémonique) qui, utilisée en combinaison avec la touche ALT, permet d'ouvrir un menu de la barre de menu. La touche d'accès rapide correspond à l'un des caractères de l'étiquette de la commande. Par exemple, appuyer sur ALT+F active un menu dont l'étiquette est *Fichier* et dont la touche d'accès rapide est F. Généralement, les touches d'accès rapide ne font pas de distinction entre les majuscules et les minuscules.

Vous pouvez attribuer des touches d'accès rapide à des commandes en suivant les lignes directrices suivantes (dans cet ordre) :

1. Utilisez la première lettre de l'étiquette de la commande, à moins qu'une autre lettre offre une meilleure association mnémonique.
2. Utilisez une consonne caractéristique dans l'étiquette.
3. Utilisez une voyelle dans l'étiquette.

Évitez les attributions où l'indication visuelle de la touche d'accès rapide ne peut être distinguée du caractère (ex. : i ou L). Évitez aussi d'utiliser des caractères habituellement employés pour une fonction répandue. Par exemple, lorsque vous incluez un bouton *Appliquer*, réservez le A comme touche d'accès rapide à ce bouton. De plus, n'attribuez pas de touches d'accès rapide aux commandes OK et *Annuler*, lorsqu'elles sont respectivement jumelées aux touches ENTRÉE et ECHAP (ESC).

Les touches d'accès rapide qui sont attribuées à plus d'une commande dans un même contexte (tel qu'un menu ou un sous-menu) donnent accès à la première commande. Selon la commande, si l'utilisateur appuie sur la touche une seconde fois, elle pourrait, ou non, donner accès à la seconde commande lui ayant été attribuée. Par conséquent, l'attribution d'une touche d'accès rapide devrait être unique dans un même contexte d'interaction.

2.2.3 Touches de mode

Les touches de mode changent les actions d'autres touches ou encore de dispositifs d'entrée de données. On distingue les touches de mode *bascule (toggle)* et *modification*. Une touche de mode *bascule* active ou désactive un mode précis chaque fois qu'elle est enfoncée. Par exemple, appuyer sur la touche CAPS LOCKS active les touches de texte en majuscules ; appuyer sur la touche NUM LOCKS permet de basculer entre les entrées numériques et les entrées de navigation directionnelle pour le clavier numérique.

Tout comme les touches de mode *bascule*, les touches de mode *modification* changent l'action de certaines entrées. Par contre, le mode activé (ou désactivé) demeure actif (ou inactif) seulement lorsque la touche de mode *modification* est maintenue enfoncée. Les touches de mode *modification* comprennent les touches MAJ (SHIFT), ALT et CTRL. Un tel mode à *effet-ressort* est souvent préférable à un mode *verrouillé* parce qu'il exige l'activation par l'utilisateur, ce qui rend le choix de ce mode délibéré et permet à l'utilisateur de l'annuler aisément en relâchant la touche.

Évitez d'employer une multitude de clés de modification en guise d'accès principal pour des opérations élémentaires. Il est possible que dans certains contextes, tels que des environnements propres à l'entrée de données à l'aide d'un crayon, le clavier ne soit pas disponible. Par conséquent, utilisez des actions basées sur des touches de mode *modification* seulement pour accélérer l'accès aux opérations auxquelles il est possible d'accéder par un autre moyen dans l'interface.

2.2.4 Raccourci au clavier

Les *raccourcis au clavier* (également appelés *touches d'accélération*) sont des touches ou des combinaisons de touches qui permettent, lorsqu'elles sont enfoncées, l'accès rapide à des opérations couramment effectuées. Les combinaisons CTRL+*Lettre* et les touches de fonction (F1 à F12) sont habituellement le meilleur choix pour des raccourcis au clavier. Par définition, un raccourci au clavier est l'équivalent clavier d'une fonctionnalité à laquelle il est possible d'accéder autrement dans l'interface. Évitez d'employer un raccourci au clavier comme seul moyen d'accès à une opération précise.

Lors de l'assignation de raccourcis au clavier, suivez ces lignes directrices :

- Lorsque cela est possible, n'utilisez qu'une seule touche alphanumérique (ex. : MAJ+A plutôt que MAJ+A+B).
- Ne faites pas de distinction entre les combinaisons de majuscules et de minuscules.

- Utilisez la combinaison MAJ+*Touche* pour des actions qui prolongent ou sont complémentaires des actions d'une touche ou d'une combinaison de touches qui n'utilisent pas MAJ. Par exemple, la combinaison ALT+TAB bascule entre les fenêtres empilées du bas vers le haut. La combinaison MAJ+ALT+TAB bascule dans l'ordre inverse. Toutefois, évitez les combinaisons de touches MAJ+*Lettre* car l'effet de la touche MAJ peut varier pour certains claviers internationaux.
- Utilisez la combinaison CTRL+*Touche* pour des actions dont l'effet se fait sentir à plus grande échelle. Par exemple, dans un contexte d'édition de texte, la touche HOME déplace le curseur au début d'une ligne ; la combinaison CTRL+HOME déplace le curseur au début du texte. Utilisez la combinaison CTRL+*Lettre* pour accéder aux commandes auxquelles une touche lettre est assignée, telle que CTRL+G pour gras. Toutefois, retenez que de telles assignations ne sont significatives que pour ceux qui parlent la langue employée (le français, dans le cas qui nous occupe).
- Évitez les combinaisons de touches ALT+*Touche* car elles peuvent entrer en conflit avec les accès standard aux menus et aux commandes. Les combinaisons ALT+*Touche*, telles que ALT+TAB, ALT+ECHAP et ALT+BARRE D'ESPACE, sont réservées au système d'exploitation. Les combinaisons ALT+*Chiffre* entrent des caractères spéciaux à l'écran.
- Évitez d'assigner les raccourcis au clavier décrits dans ce guide à d'autres opérations de votre logiciel. Par exemple, si la combinaison CTRL+C est le raccourci pour la commande *Copier* et que votre application comporte cette commande, n'assignez par la combinaison CTRL+C à une autre commande.
- Vous pouvez offrir à l'utilisateur la possibilité de modifier l'assignation des raccourcis au clavier dans votre application. Dans l'optique des applications prévues pour le RSSS, il serait sans doute préférable d'évaluer attentivement l'impact de cette option : la cohérence des applications à travers l'ensemble du réseau est un facteur de conception important.
- Utilisez la touche ECHAP pour interrompre l'exécution d'une commande ou pour annuler une opération de manipulation directe. De plus, cette touche est généralement considérée comme un raccourci au clavier pour la commande *Annuler*.

2.3 AUTRES DISPOSITIFS D'INTERACTION

Le système supporte également d'autres dispositifs d'interaction, lorsque les pilotes (*drivers*) appropriés sont installés. Parmi les autres dispositifs d'entrée de données répandus, on retrouve le *stylet (pen input)* et l'*écran tactile (touch screen)*. À l'aide du stylet ou du doigt, on peut *Taper (tap)* l'écran pour pointer, sélectionner et activer des objets de l'interface. De plus, le stylet permet également d'écrire sur l'écran avec de l'encre électronique, de manière similaire à l'écriture manuscrite sur du papier.

2.3.1 Reconnaissance de la parole

Il n'existe pas de règle stylistique officielle propre à ce dispositif. Veuillez consulter les éléments génériques du *Guide d'ergonomie* et du *Guide de style*.

2.3.2 Écran tactile

Il n'existe pas de règle stylistique officielle propre à ce dispositif. Veuillez consulter les éléments génériques du *Guide d'ergonomie* et du *Guide de style*.

2.3.3 Reconnaissance optique des caractères

Il n'existe pas de règle stylistique officielle propre à ce dispositif. Veuillez consulter les éléments génériques du *Guide d'ergonomie* et du *Guide de style*.

2.3.4 Ardoise électronique

Le contenu de ce chapitre provient du guide de style de Microsoft pour la version 3.1 de Windows [WIG92]. Il a été complété par des indications supplémentaires et des exemples provenant du guide de style de Windows 95 [WIG95].

2.3.4.1 L'entrée de données avec le stylet

Les systèmes équipés d'un logiciel de gestion du stylet sur Windows supportent l'entrée de données par le contact du stylet sur la surface d'un écran ou d'une tablette et dans certains cas, du doigt.

Le stylet peut être utilisé à la fois pour le pointage et l'écriture, en fonction de sa position spatiale. Par exemple, si l'on déplace le stylet au-dessus des menus et de la plupart des objets de contrôle, il agit comme dispositif de pointage. Grâce aux possibilités de pointage avec le stylet, l'utilisateur peut effectuer plusieurs opérations équivalentes à celles de la souris. Lorsque le curseur se trouve au-dessus d'une zone d'entrée de texte ou de dessin, le stylet agit comme dispositif d'entrée de données : le pointeur prend alors l'apparence d'un stylo et produit à l'écran une sorte « d'encre » électronique qui permet de tracer des formes à l'écran.

Le stylet peut conserver sa fonctionnalité de dispositif de pointage même dans les contextes où il devrait normalement agir comme dispositif d'écriture ou de dessin. Par exemple, il est possible d'utiliser une variable temporelle pour différencier les deux opérations : si l'utilisateur maintient une pression du stylet au même endroit sur la surface de l'écran pendant une durée prédéterminée, une action différente peut être inférée. Cependant, cette méthode est peu fiable et inefficace pour les opérations multiples. Il est donc préférable de recourir à des boutons sur une barre d'outils pour passer d'un mode à l'autre.

Il est également possible de fournir à l'utilisateur l'accès à d'autres opérations par le biais d'un *opérateur d'interaction (action handle)*. Un *opérateur d'interaction* consiste en une fenêtre spéciale utilisée pour la sélection d'opérations de manipulation directe ou de menus autonomes (*pop-up menus*).

Les actions suivantes sont des mouvements fondamentaux définis pour l'interaction avec le stylet :

Tableau 2 : Mouvements fondamentaux du stylet

Peser	Positionner et appuyer l'extrémité du stylet sur la surface d'entrée de données. Une pression du stylet équivaut à pointer avec la souris et correspond à une action particulière du stylet.
Taper* (<i>Tap-ping</i>)	Frapper l'extrémité du stylet sur la surface d'entrée de données puis soulever sans déplacer le stylet. Généralement, cette action est équivalente au cliquage du bouton de la souris. Cette action permet de sélectionner un objet, de placer le curseur pour l'insertion de texte ou d'activer un bouton.
Double-Taper	Frapper et soulever rapidement l'extrémité du stylet à deux reprises successives. Cette action est équivalente au double-cliquage de la souris.
Glisser-Déposer	Appuyer l'extrémité du stylet sur la surface d'entrée et maintenir la pression tout en déplaçant le stylet.

* Tapping : dans le *Guide de style* de Windows95, cette expression est traduite par *Toucher*. Dans le cadre de ce guide, nous avons choisi le terme *Taper*.

La probabilité que l'utilisateur déplace le stylet pendant l'exécution du *Double-Taper* (*double-tap*) est plus élevée qu'avec la souris. Il est donc préférable d'agrandir la zone sensible (*hot zone*) pour permettre une meilleure interprétation de l'action.

Certains stylets comportent un bouton, placé sur le cylindre du crayon, qui peut être enfoncé. Pour les stylets qui supportent ce bouton, les actions *Taper* ou *Glisser-Déposer* effectuées en maintenant le bouton du stylet enfoncé sont équivalentes aux actions *Cliquer* et *Glisser-Déposer* effectuées avec la souris en maintenant le bouton 2 enfoncé. Étant donné que les stylets ne disposent pas tous de ce bouton, toutes les actions qui relèvent de celui-ci devraient aussi être supportées par une autre technique d'interaction.

La *proximité* est la capacité de détecter la position du stylet sans que celui-ci ne touche la surface. Même si Windows offre cette fonctionnalité, il faut éviter d'y recourir en tant que moyen exclusif d'accès aux fonctions de base, parce que les stylets ne supportent pas nécessairement cette fonction.

2.3.4.2 Les pointeurs

Pour les ardoises électroniques, tout comme pour la souris, les pointeurs jouent un rôle important pour indiquer visuellement la position du curseur à l'écran, ainsi que la nature de l'interaction. Les pointeurs permettent à l'utilisateur de sélectionner rapidement de petites cibles à l'écran. Les changements d'apparence du pointeur fournissent un retour d'information sur l'action qui est supportée par l'objet au-dessus duquel se trouve le stylet. Par exemple, lorsque le stylet glisse au-dessus d'une bordure pouvant être redimensionnée, le pointeur prend l'apparence d'une flèche. Autant que possible, il faut inclure ce type de retour d'information dans les applications afin d'aider l'utilisateur à mieux comprendre quelles sont les actions supportées.

Deux pointeurs sont communément utilisés : la flèche et le crayon.



Sert à pointer, sélectionner, déplacer et redimensionner

Sert à écrire et dessiner

Lorsque l'écran est également la surface d'entrée de données et que le pointeur peut être masqué par le stylet ou la main de l'utilisateur, il est souhaitable de considérer, en plus du retour d'information additionnel, les barres d'outils (*toolbar button states*) ou les barres d'état (*status bar*) pour indiquer l'état du stylet.

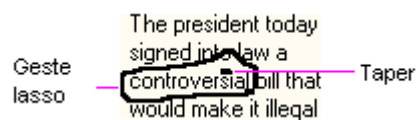
2.3.4.3 La sélection avec le stylet

Lorsque le stylet est utilisé comme dispositif de pointage, les techniques de sélection peuvent être les mêmes qu'avec la souris. Les interfaces de stylet standard supportent également la sélection de texte en utilisant un opérateur de sélection de stylet (*special pen selection handle*).

Dans certains contextes spécifiques, il est également possible d'utiliser la technique du *Peser-Glisser-Déposer* ou la technique du *Lasso-Taper* pour la sélection d'un objet ou d'un ensemble d'objets. La technique du *Lasso-Taper* consiste à effectuer un mouvement circulaire autour de l'objet et ensuite taper au-dessus du même objet. Pour la technique du *Peser-Glisser-Déposer*, le passage d'un mode de sélection à l'autre s'effectue lorsque l'utilisateur maintient une pression de l'extrémité du stylet au même endroit pendant une durée prédéterminée. L'utilisateur peut alors déplacer le stylet pour faire une sélection.

Il est important d'éviter de recourir à ces techniques dans les cas où elles entrent en conflit avec les opérations de base telles que la manipulation directe.

Figure 3 : Geste Lasso-Taper



Dans une situation de manipulation de texte, la sélection est fonction de l'étendue du geste du *lasso* et de la granularité des éléments de texte (caractères-mots-paragraphe). Par exemple, si l'utilisateur esquisse un lasso autour d'un seul caractère, sélectionner uniquement ce caractère. Si l'utilisateur esquisse un lasso autour de plusieurs caractères d'un mot, sélectionner le mot en entier. Si le mouvement englobe plusieurs mots, sélectionner la chaîne de mots incluse dans le geste. Ces règles permettent de réduire la précision gestuelle exigée de l'utilisateur.

2.3.4.4 L'utilisation des gestes avec le stylet (*pen gestures*)

Lorsque le stylet est utilisé pour l'écriture, certains dessins à main levée (*ink patterns*) sont interprétés comme des gestes (*gestures*). Ces gestes font référence à une opération particulière : effacer du texte, produire un caractère non imprimable, comme un retour de chariot ou un espace de tabulation (*tab*). Par exemple, le geste *X encerclé* équivaut à la commande *Couper*. Lorsque le système a interprété le geste, le dessin à main levée qui le représente est retiré de l'affichage.

Tous les gestes sont entourés d'un cercle pour les distinguer des caractères ordinaires. La plupart des gestes fonctionnent par positionnement, c'est-à-dire que leur action porte sur l'objet sur lequel ils sont esquissés. La détermination de la position d'un geste spécifique dépend soit de la zone qu'il inclut, soit d'un point précis du geste, c'est-à-dire son point d'impact (*hot spot*).

Les gestes du stylet ne peuvent habituellement pas être combinés avec d'autres dessins à main levée dans une même séquence de reconnaissance. Par exemple, un utilisateur ne peut pas dessiner quelques caractères, enchaîner immédiatement avec un geste et ensuite poursuivre en dessinant d'autres caractères.

La rapidité des commandes gestuelles est un des avantages clés du stylet. Par contre, il ne faut pas dépendre uniquement d'un geste pour l'exécution d'une commande puisque cela requiert un effort de mémorisation de la part de l'utilisateur. Il faut considérer les gestes comme un raccourci pour des opérations adéquatement supportées ailleurs dans l'interface comme dans les menus ou les boutons de commande. Il est possible, lorsque les extensions du stylet sont installées, de placer une icône (*bitmap*) du geste à côté de la commande correspondante (en lieu et place du raccourci au clavier), afin d'aider l'utilisateur à apprendre des gestes particuliers.

De plus, il faut éviter d'utiliser des gestes lorsque ces derniers interfèrent avec des fonctions répandues ou lorsqu'ils commandent des opérations en parallèle avec d'autres dispositifs d'entrée tels que la souris ou le clavier. Par exemple, même s'il était possible d'utiliser un geste pour faire défiler une liste, ce geste pourrait interférer avec l'action de base fréquemment employée qui consiste à sélectionner un élément dans la liste. Une option intéressante consisterait à offrir une zone de texte où l'utilisateur pourrait écrire : la liste défilerait en fonction des caractères insérés.

2.3.4.4.1 Les gestes d'édition avec le stylet (*pen editing gestures*)

Le stylet, lorsqu'il est utilisé comme outil d'écriture, permet le recours aux techniques d'édition mises au point pour la souris. Les gestes fonctionnent par positionnement (sauf pour la commande *Défaire*), agissant sur les objets sur lesquels ils sont esquissés. La liste des gestes est présentée dans le tableau ci-dessous. Lorsque l'utilisateur esquisse un geste sur un objet non sélectionné, le geste agit sur cet objet, même s'il existe un autre objet sélectionné dans la même zone. Par contre, si l'utilisateur esquisse un geste sur une zone contenant des objets, où certains sont sélectionnés et d'autres non, ce geste ne s'applique qu'aux objets sélectionnés. Lorsque le geste est esquissé sur un seul des éléments de la sélection, alors il s'applique à la sélection entière. Si le geste est esquissé sur un espace vide (à l'arrière-plan – *background*), il s'applique à tout objet sélectionné contenu dans la région effective de la sélection. S'il n'y a aucune sélection dans cette région, alors le geste n'a aucun effet.

Pour la plupart des gestes, le point d'impact détermine spécifiquement l'objet sur lequel le geste s'applique. Si le point d'impact peut se trouver sur n'importe quelle partie d'une sélection, alors le geste s'applique sur toute la sélection.

Figure 4 : Les gestes d'édition avec le stylet

Signe	Nom	Opération
	Crochet encerclé	Pour les objets de type texte, affiche les outils d'écriture de la fenêtre d'édition. Pour les autres objets, affiche les propriétés.
	c encerclé	Copier
	d encerclé	Effacer (ou Supprimer, selon le contexte)
	m encerclé	Menu
	n encerclé	Nouvelle ligne
	p encerclé	Coller
	s encerclé	Insérer un espace
	t encerclé	Insérer une tabulation
	u encerclé	Défaire
	x encerclé	Couper
	Caret encerclé	Insérer du texte

Ces gestes peuvent être spécifiques en fonction des différentes versions internationales. Par exemple, dans la version japonaise, le *k encerclé* est utilisé pour la conversion du Kana au Kanji.

2.3.4.5 La reconnaissance de l'écriture

La reconnaissance est l'interprétation du trait du crayon en fonction d'une signification standardisée. Il faut considérer la reconnaissance comme un moyen pour arriver à une fin et non comme une fin en soi. Ne pas faire appel à la reconnaissance de l'écriture si cela n'est pas utile au contexte d'utilisation ou si ce n'est pas la meilleure interaction. Par exemple, pour obtenir une date de votre utilisateur, il est plus efficace de fournir un contrôle qui lui permet de sélectionner une date plutôt que de lui demander de l'écrire afin de la faire reconnaître par votre application.

Une reconnaissance fidèle est difficile à atteindre, mais il est possible d'améliorer de manière importante l'interaction de reconnaissance en offrant des moyens simples et rapides de corriger des erreurs. Par exemple, si l'on offre à l'utilisateur la possibilité d'écrire par-dessus les caractères ou d'effectuer un choix parmi des possibilités diverses, il sera moins frustré et trouvera la reconnaissance plus utile. Il est

également possible d'améliorer la reconnaissance en tirant profit du contexte et des contraintes. Par exemple, il est possible de limiter l'entrée de données d'un champ à des données numériques.

2.3.4.6 Le stockage d'encre

Dans certains cas – les signatures, par exemple – la reconnaissance de l'écriture n'est pas obligatoire : le stockage d'encre devient alors une représentation d'information suffisante. L'encre est un type de données standard supporté par le *Presse-Papiers (clipboard)*. Considérez le stockage d'encre comme moyen d'entrée de données lorsque votre application permet l'entrée de texte, sauf dans les cas où la représentation de ces données doit être interprétée pour d'autres opérations, telles que le tri ou la recherche d'information.

2.3.4.7 Positionnement du curseur

Le positionnement, c'est-à-dire l'emplacement où sera effectuée l'entrée de données, est un facteur de conception important pour les systèmes utilisant le stylet. Par exemple, si l'utilisateur déplace le stylet au-dessus de plusieurs objets, lesquels seront modifiés ? Voici les principes associés au positionnement du geste sur les objets :

- Si l'utilisateur esquisse le geste (*draws the gesture*) sur n'importe quelle partie d'une sélection, appliquer le geste sur cette sélection.
- Si l'utilisateur esquisse le geste sur un objet qui n'est pas sélectionné, sélectionner cet objet et appliquer le geste sur cet objet.
- Si l'utilisateur esquisse le geste sans qu'il y ait d'objet situé directement au-dessous, mais qu'il y a une sélection, appliquer le geste sur cette sélection.

Si aucun de ces principes ne s'applique, ignorer le geste.

Pour l'écriture manuscrite, il est également possible d'utiliser le contexte pour déterminer la zone d'entrée. La figure suivante démontre comment la proximité du texte par rapport à la boîte de texte détermine la destination du texte écrit. En effet, même si le texte manuscrit dépasse les limites des zones de texte, le système est en mesure de déterminer à quelle zone de texte l'écriture doit être associée.

Figure 5 : Positionnement de l'entrée d'écriture manuscrite



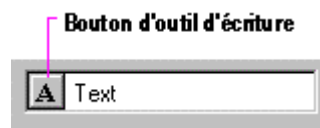
L'extension stylet du système fournit un support de base pour le positionnement, mais votre application peut fournir un support additionnel. Par exemple, l'application peut présenter un rectangle plus large que le contrôle par défaut. De plus, comme votre application connaît le type de données associé à un champ donné, elle peut utiliser cette information pour mieux interpréter le positionnement des données.

2.3.4.8 Techniques d'édition spécifiques du stylet

2.3.4.8.1 *L'édition dans les contrôles permettant l'usage du stylet (Editing in Pen-Enabled Controls)*

Si un stylet est installé, le système fournit automatiquement une interface spéciale, appelée *outil d'écriture*. Cet outil facilite l'édition de texte, améliore la précision de la reconnaissance et la correction des erreurs. L'interface de l'outil d'écriture ajoute un bouton aux contrôles de texte standard. Ce nouveau contrôle réduit la zone visible d'une zone de texte : il faut prendre cet élément en considération lors de la détermination des dimensions des boutons de contrôle.

Figure 6 : Zone de texte standard avec bouton d'outil d'écriture



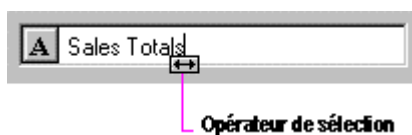
La figure suivante montre comment ajouter le support de l'outil d'écriture pour tout besoin particulier de votre application, tel qu'une zone de texte multiligne.

Figure 7 : Ajout du bouton d'outil d'écriture



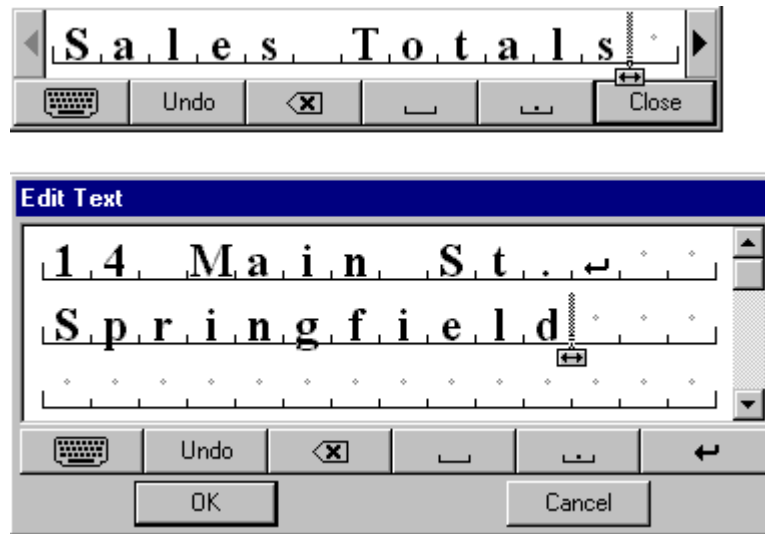
Lorsqu'un contrôle de zone de texte est le centre d'attention, un opérateur de sélection (*selection handle*) apparaît. L'utilisateur peut faire glisser cet opérateur pour faire une sélection.

Figure 8 : Boîte de texte affichant un opérateur de sélection



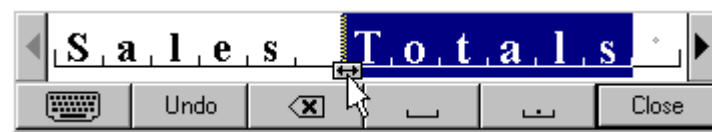
Lorsqu'on tape sur le bouton de commande de l'outil d'écriture avec le stylet (ou le cliquage de la souris), une fenêtre spéciale d'édition de texte apparaît. Dans cette fenêtre, l'utilisateur peut écrire du texte, qui est alors déchiffré automatiquement.

Figure 9 : Fenêtres d'outil d'écriture uniligne et multiligne



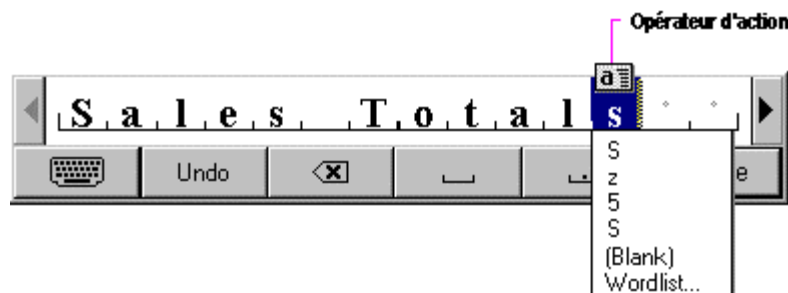
Dans la fenêtre de l'outil d'écriture, chaque caractère est affiché à l'intérieur d'une cellule spéciale. Si l'utilisateur sélectionne du texte dans la zone de texte originale, la fenêtre de l'outil d'écriture reflète cette sélection. L'utilisateur peut déplacer la sélection vers un nouveau point d'insertion en tapant entre les caractères.

Figure 10 : Sélection de texte avec l'opérateur de sélection



L'utilisateur peut sélectionner un seul caractère dans sa cellule en tapant, ou encore sélectionner un mot complet en double-tapant. Lorsque l'utilisateur tape sur un seul caractère, un opérateur d'interaction affiche une liste de caractères alternatifs.

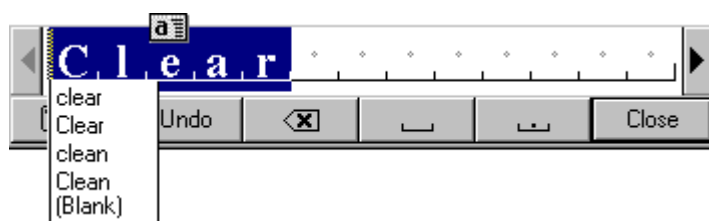
Figure 11 : Opérateur d'action avec liste d'options



Lorsque l'utilisateur choisit un autre caractère dans cette liste, ce dernier remplace l'ancien caractère dans le texte puis la liste est retirée de l'affichage. Écrire par-dessus un caractère ou taper ailleurs fait aussi disparaître la liste. Le nouveau caractère remplace l'ancien et déplace la sélection vers un point d'insertion situé à la gauche du nouveau caractère.

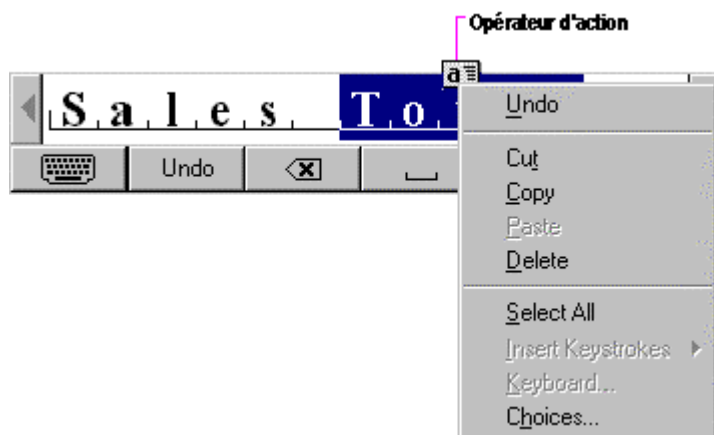
La liste contient aussi un élément étiqueté *Wordlist*. Lorsque l'utilisateur sélectionne cet élément, le mot contenant le caractère devient sélectionné et une liste de mots alternatifs est affichée. La liste apparaît aussi lorsque l'utilisateur sélectionne un mot complet en double-tapant. Le choix d'un élément de la liste remplace le mot sélectionné.

Figure 12 : Affichage d'une liste d'options par tape



Lorsqu'il y a une sélection dans la fenêtre, un opérateur d'interaction apparaît. L'utilisateur peut l'utiliser pour effectuer d'autres opérations sur les éléments sélectionnés. Par exemple, il est possible de déplacer ou copier la sélection par glissement ou encore d'accéder à son menu autonome en tapant sur l'opérateur d'action.

Figure 13 : Accès au menu autonome par tape

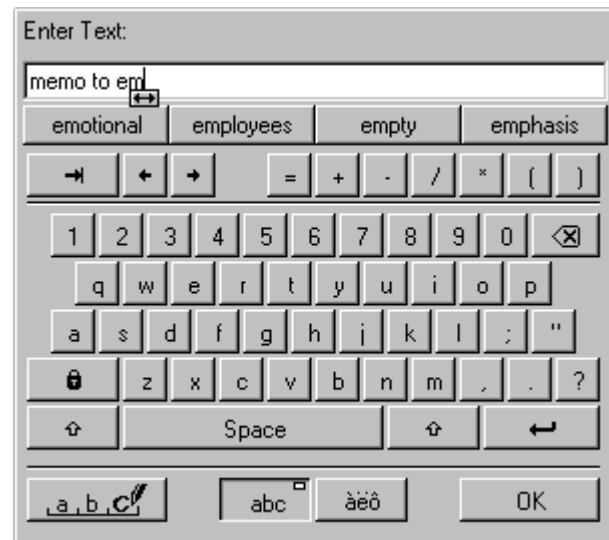


Les boutons de commande de la fenêtre de l'outil d'écriture sont utilisés pour le défilement du texte aussi bien que pour les fonctions répandues comme : *Défaire*, *Rappel de chariot* (*Backspace*), *Insérer un espace*, *Insérer un point* et *Fermer* (pour fermer la fenêtre de texte).

La fenêtre de l'outil d'écriture offre aussi un bouton permettant l'affichage d'un clavier en guise d'alternative à l'écriture de caractères avec le stylet. L'utilisateur tape sur le bouton de commande qui contient

une icône représentant un clavier : la fenêtre de l'outil d'écriture est remplacée par une fenêtre autonome présentant un outil clavier à l'écran.

Figure 14 : Fenêtre autonome de l'outil clavier à l'écran



L'outil d'écriture garde en mémoire son usage précédent, soit *entrée de texte* ou *clavier à l'écran* : lors d'un usage ultérieur, il apparaît dans la fenêtre d'édition appropriée. De plus, notez que lorsque l'utilisateur fait apparaître une fenêtre d'outil d'écriture, cette dernière devient le centre d'attention. Il faut faire en sorte que le déplacement du centre d'attention ne devienne pas une indication que l'interaction avec un champ donné est terminée ou encore que toute édition de texte s'effectue directement dans une zone de texte.

3. TECHNIQUES D'INTERACTION

Cette section couvre les techniques d'interaction de base, telles que la manipulation, la création et l'édition de données. Une bonne partie de ces techniques sont basées sur un paradigme d'action sur les objets dans lequel l'utilisateur doit d'abord identifier un objet et ensuite choisir une action à lui appliquer. En implantant ces techniques de façon cohérente, vous permettez aux utilisateurs de transférer leurs connaissances vers de nouvelles tâches.

Lorsque cela est possible, rendez disponibles ces techniques d'interaction de base pour le clavier, la souris ou encore d'autres dispositifs d'entrée tels que le stylet ou l'écran tactile. Lorsque vous ajoutez ou que vous modifiez ces techniques pour un dispositif particulier, prenez en considération la manière dont la nouvelle caractéristique ou la nouvelle fonction pourra être réalisée par les autres dispositifs d'interaction. Une technique pour un dispositif donné n'a pas à être identique à celles utilisées pour les autres dispositifs. Concevez plutôt les techniques de manière à mettre en valeur les qualités d'un dispositif en particulier. De plus, facilitez le transfert d'un dispositif à un autre pour l'utilisateur de manière à ce que l'interaction amorcée avec un dispositif puisse être complétée par un autre.

3.1 TRANSFORMATIONS

Une *transformation* (*transaction*) est une unité de changement apporté à un objet. Par exemple, un changement au format d'un caractère (le mettre en gras) ou à l'emplacement d'un fichier (déplacer un fichier à l'aide des commandes *Couper* et *Coller*). La granularité d'une transformation peut aller du résultat d'une opération unique au résultat d'un ensemble d'opérations.

Note : Le terme *transformation* est une traduction libre du terme anglais *transaction*. Le terme français a été choisi de manière à éviter la confusion qui pourrait survenir avec le terme *transaction* utilisé à l'intérieur du RSSS pour décrire les échanges entre les intervenants et les usagers. Bien que la définition donnée ci-dessus ne semble pas intuitive, elle correspond à la définition donnée dans [WIG95]. Nous l'utiliserons telle quelle dans le présent document.

Dans un modèle idéal, les transformations sont exécutées immédiatement et il y a possibilité de « défaire » les transformations. Parce qu'il y a des circonstances où cette façon de faire n'est pas pratique, des conventions d'interaction spécifiques ont été établies pour l'exécution des transformations. S'il y a encore des transformations en attente de traitement dans une fenêtre que l'on veut fermer, il faut toujours demander à l'utilisateur s'il souhaite exécuter ou éliminer ces transformations.

Les transformations peuvent être exécutées à plusieurs niveaux : une transformation exécutée à un niveau donné n'implique pas nécessairement un changement permanent. Par exemple, l'utilisateur peut changer la police de caractère d'un texte sélectionné, mais l'enregistrement (*save*) du fichier du document pourrait être exigé avant que le changement soit rendu permanent.

Utilisez les commandes suivantes pour exécuter des transformations relatives aux fichiers :

Commande	Fonction
Enregistrer	Sauvegarde sur le disque toutes les éditions effectuées depuis le dernier enregistrement (ou lecture) et commence une nouvelle séance d'édition.
Enregistrer sous	Sauvegarde le fichier sous un nouveau nom et commence une nouvelle séance d'édition.
Fermer	Invite l'utilisateur à enregistrer toutes les nouvelles éditions effectuées depuis le dernier enregistrement (ou lecture). S'il y a confirmation, les éditions sont enregistrées et la fenêtre est fermée.

Note : Utilisez la commande *Enregistrer* lorsque l'exécution de transformations de fichiers s'applique pour un fichier entier, tel qu'un document, et est effectuée à un moment précis.

Pour les niveaux plus spécifiques, vous pouvez utiliser les commandes suivantes pour le traitement des transformations de manipulations les plus répandues.

Commande	Fonction
Répéter	Reproduit la dernière transformation exécutée.
Défaire	Annule la dernière transformation exécutée, ou encore celle spécifiée par l'utilisateur, et ramène l'interface à son état initial.
Rétablir	Reproduit la dernière transformation annulée, ou encore celle spécifiée par l'utilisateur.
OK	Exécute les transformations en attente de traitement et ferme la fenêtre.
Appliquer	Exécute les transformations en attente de traitement mais ne ferme pas la fenêtre.
Annuler	Élimine les transformations en cours de traitement et ferme la fenêtre.

Note : Plusieurs applications recourent à la commande *Annuler* pour obtenir le comportement décrit pour la commande *Défaire*. La commande *Annuler* est d'usage plus répandu dans la plupart des interfaces.

Les commandes suivantes sont recommandées pour le traitement des transformations opérationnelles :

Commande	Fonction
Pause	Suspend temporairement l'exécution d'une opération.
Reprendre	Reprend une opération suspendue.
Arrêter	Met fin à une opération.

Note : Bien qu'il soit possible d'utiliser la commande *Annuler* pour mettre fin à une opération, *Annuler* implique que l'objet de l'opération sera remis à son état initial précédant l'activation de la commande.

3.2 OPÉRATIONS DE TRANSFERT

Les opérations de transfert sont des opérations qui impliquent (ou peuvent être dérivées de) les opérations *Déplacer* et *Copier*. Par exemple, l'impression d'un objet est une forme d'opération de transfert puisqu'elle peut être définie comme étant la copie d'un objet vers l'imprimante.

Les opérations de transfert sont constituées de trois composantes : l'objet à déplacer, la destination et l'opération à exécuter. On peut les définir implicitement ou explicitement, selon la technique d'interaction utilisée. L'opération définie par un transfert est déterminée par sa destination. Puisqu'il est possible qu'un transfert fasse l'objet de plusieurs interprétations, vous pouvez définir une opération par défaut ainsi que d'autres opérations optimales, en fonction des informations fournies par la source du transfert, ainsi que par la compatibilité et les capacités de la destination. Par exemple, tenter de déplacer un objet vers un contenant peut amener l'un des résultats suivants :

- rejet de l'objet ;
- acceptation de l'objet ;
- acceptation d'un sous-ensemble ou encore d'une forme modifiée de l'objet.

La plupart des transferts sont basés sur l'une des trois opérations fondamentales suivantes :

Opération	Description
Déplacer	Donne une nouvelle position à l'objet sélectionné. Parce que l'identité de base de l'objet déplacé n'est pas altérée, cette opération n'est pas la même que celle qui consiste à copier un objet et à détruire l'original.
Copier	Duplique un objet. L'objet résultant de cette opération est entièrement indépendant de l'original. <i>Copier</i> ne produit pas toujours un clone identique : certaines des propriétés du duplicata peuvent différer de celles de l'original. Par exemple, le duplicata d'un objet peut avoir un nom et une date de création différents. De même, si certaines composantes de l'objet limitent la duplication, seules les composantes sans restrictions pourront être reproduites.

Dans l'interface Windows, il existe deux méthodes pour effectuer les transferts : la méthode des commandes et la méthode de manipulation directe.

3.2.1 Méthode des commandes

Pour déplacer des objets, cette méthode utilise les commandes *Couper*, *Copier* et *Coller*. Placez ces commandes dans le menu déroulant *Edition*, ainsi que dans le menu autonome associé à un objet. Vous pouvez également employer des boutons de la barre d'outils pour activer ces commandes.

Pour transférer un objet, l'utilisateur doit :

1. faire une sélection ;
2. choisir la commande *Couper* ou *Copier* ;
3. se déplacer à l'emplacement destination et indiquer le point d'insertion, si nécessaire ;
4. choisir l'une des opérations *Coller*.

La commande *Couper* enlève la sélection et la transfère (ou transfère sa référence) au *Presse-Papiers*. La commande *Copier* duplique la sélection (ou sa référence) et la transfère au *Presse-Papiers*. La commande *Coller* complète l'opération de transfert vers la destination. Par exemple, lorsque l'utilisateur choisit *Couper* et *Coller*, enlevez la sélection de sa source et relocalisez-la dans sa destination. Pour *Copier* et *Coller*, insérez un duplicata indépendant de la sélection à l'emplacement destination tout en laissant l'original intact. Lorsque l'utilisateur choisit la commande *Copier* et *Coller le raccourci*, insérez un objet à la destination relié à la source.

La commande *Coller* se présente sous plusieurs formes. Choisissez la forme qui indique comment l'objet sera déplacé vers la destination. Utilisez la commande *Coller* pour indiquer que l'objet sera déplacé en conservant sa forme première. Vous pouvez également employer des formes alternatives de la commande *Coller* pour d'autres types de transferts, selon la forme générale :

Coller [*nom type*] [**sous** *nom type* | **vers** *nom objet*]

Dans l'exemple « *Coller Cellule sous Tableau Word* », le nom du type est *Cellule* et le nom du type de conversion est *Tableau Word*.

La liste suivante présente les différentes formes de la commande *Coller* :

Commande	Fonction
Coller	Insère l'objet situé dans le <i>Presse-Papiers</i> en conservant sa forme première.
Copier [<i>nom type</i>]	Insère l'objet situé dans le <i>Presse-Papiers</i> en tant qu'objet OLE incorporé. Cet objet peut être activé directement dans la destination.
Coller [<i>nom type</i>] en tant qu'icône	Insère l'objet situé dans le <i>Presse-Papiers</i> en tant qu'objet OLE incorporé. Cet objet est affiché en tant qu'icône (format <i>Icon</i>).
Collage spécial	Présente une boîte de dialogue qui donne à l'utilisateur le contrôle explicite sur la manière d'insérer l'objet dans le <i>Presse-Papiers</i> .

Utilisez le contexte de la destination pour déterminer quelle(s) forme(s) de la commande *Coller* inclure : ce choix s'effectue en fonction des options que la commande peut offrir à l'utilisateur, selon les différentes formes de l'objet rendues disponibles par la source. Ce choix dépend également de la nature ou du but servi par la destination. Par exemple, une imprimante définit son propre contexte de transfert.

Généralement, seules les commandes *Coller* et *Collage spécial* sont utilisées. La commande *Coller* peut être modifiée dynamiquement pour refléter la forme par défaut (ou préférée) de la destination en insérant l'objet transféré sous sa forme première ou en tant qu'objet OLE incorporé. La commande *Collage spé-*

cial peut être utilisée pour le traitement de toute forme particulière de transfert. Par contre, si le contexte de la destination suggère l'accès rapide à une autre forme de transfert, vous pouvez également inclure cette commande.

Si l'utilisateur n'indique pas ses préférences à cet effet, l'application destination détermine l'option d'affichage par défaut. Pour les documents, il faut généralement afficher l'objet OLE inséré dans sa présentation usuelle.

L'exécution d'une commande *Coller* ne devrait pas avoir d'effet sur le contenu du *Presse-Papiers*. Cette caractéristique permet de coller le contenu du *Presse-Papiers* à de nombreuses reprises : les commandes *Coller* subséquentes devraient toujours produire des copies de l'original. Toutefois, une commande *Couper* ou *Copier* subséquente remplace le dernier objet situé dans le *Presse-Papiers* par un nouvel objet.

3.2.2 Méthode de manipulation directe

Dans ce type de transfert, l'utilisateur sélectionne et déplace un objet vers l'emplacement désiré à l'aide de la souris. En général, il faut éviter d'utiliser la manipulation directe comme moyen exclusif de transfert. Les meilleures interactions sont celles qui utilisent la méthode des commandes comme méthode de base pour les transferts et la manipulation directe en guise de technique de raccourci.

Il n'y a pas d'interface clavier permettant la manipulation directe des objets de l'interface. Par exemple, il n'est pas possible de sélectionner un objet à partir du clavier pour ensuite le déplacer à l'aide des touches fléchées.

Vous pouvez rendre disponibles les transferts par manipulation directe pour n'importe quel objet visible sur l'interface. L'objet (ex. : une fenêtre ou une icône) n'a pas à être la sélection active. Par exemple, un utilisateur peut déplacer un objet dans une fenêtre inactive : le déplacement de l'objet active la fenêtre. Si un objet inactif ne peut accepter un transfert par manipulation directe, cet objet (ou son contenant) doit fournir un retour d'information à l'utilisateur. Toutefois, certains transferts ne sont pas autorisés. Par exemple, il n'est pas possible de déplacer l'icône d'une application dans la fenêtre principale d'une autre application. Lorsque l'utilisateur tente un transfert interdit, le pointeur prendra alors la forme *Ne peut déplacer l'objet à cet emplacement*.

La manière dont un objet est intégré et affiché dans la destination est déterminée par le contexte de cette dernière. Un objet déplacé peut être intégré sous la forme propre à l'application (*native data*), sous une forme partielle de l'objet d'origine (comme ses propriétés), sous une forme modifiée ou encore en tant qu'objet OLE incorporé. Il vous faut déterminer si vous devez ajouter ou remplacer la sélection existante en fonction du contexte de l'opération, en vous basant sur des facteurs tels que les formats disponibles pour cet objet, le but servi par la destination et les informations fournies par l'utilisateur, par exemple l'emplacement où il a déposé l'objet ou la commande précise qu'il a utilisée. À titre d'exemple, certaines applications peuvent fournir un outil conçu spécifiquement pour copier les propriétés des objets.

Commande *Glisser-Déposer* par défaut

La commande *Glisser-Déposer* par défaut déplace un objet à l'aide du bouton 1 de la souris. L'interprétation de cette opération dépend de ce que la destination a défini comme étant l'opération par défaut. Comme pour la méthode de commande, la destination interprète l'opération en fonction de son propre contexte et des informations connues sur l'objet (et sur les formats disponibles pour ce dernier). Évitez d'utiliser une opération destructrice comme opération par défaut. Lorsque cette situation est inévitable, affichez une boîte de dialogue afin de faire confirmer le choix par l'utilisateur.

Avec cette technique, l'utilisateur peut déplacer directement les objets entre les documents définis par votre application de même que vers les ressources du système, telles que des fichiers ou une imprimante.

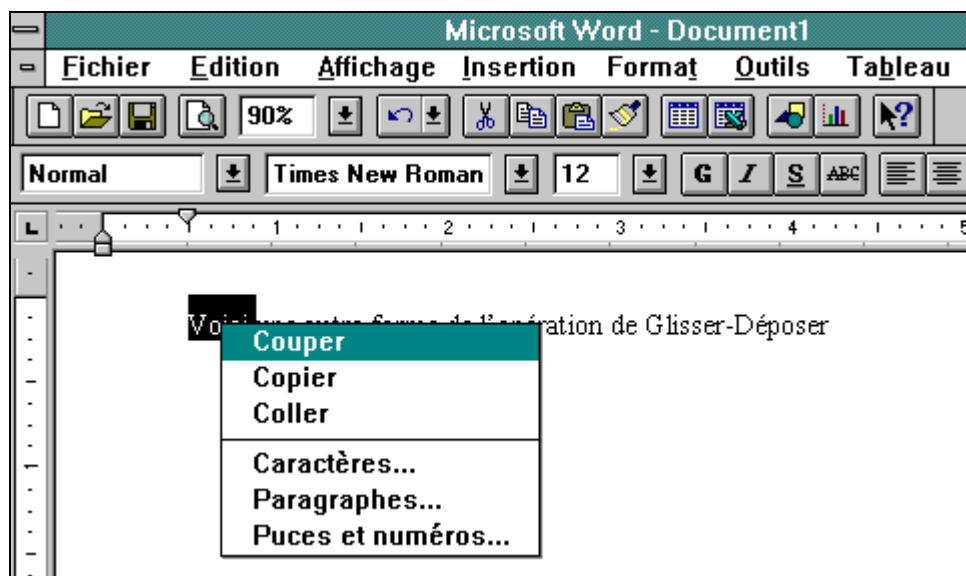
Implantez la commande *Glisser-Déposer* selon les conventions suivantes : l'utilisateur appuie sur le bouton 1 de la souris au-dessus d'un objet pour le sélectionner, déplace ensuite la souris tout en maintenant le bouton enfoncé, puis relâche le bouton pour déposer l'objet à l'emplacement destination.

L'opération par défaut la plus répandue est *Déplacer*, mais la destination (l'objet sur lequel on dépose un autre objet) peut donner une nouvelle interprétation plus adéquate de cette opération. Par conséquent, vous pouvez redéfinir une opération *Glisser-Déposer* par défaut en une autre opération générale de transfert telle que *Copier*, en une commande à destination spécifique telle que *Imprimer* ou *Envoyer*, ou même en une forme spécialisée de transfert telle que *Copier les propriétés*.

Autres formes de la commande *Glisser-Déposer*

D'autres formes de la commande *Glisser-Déposer* déplacent un objet en utilisant le bouton 2 de la souris. Dans ce cas, plutôt que d'exécuter l'opération par défaut, la destination affiche un menu autonome lorsque l'utilisateur relâche le bouton de la souris (voir la figure suivante). Le menu autonome contient l'ensemble des commandes d'achèvement de transfert appropriées qui permettent de compléter l'opération en fonction de ce que l'utilisateur veut faire.

Figure 15 : Autre forme de l'opération *Glisser-Déposer*



La destination détermine les commandes d'achèvement de transfert à inclure dans le menu autonome ; elles sont déterminées en fonction des informations sur l'objet fournies par l'emplacement source.

Ces autres formes de l'opération *Glisser-Déposer* suivent des conventions similaires à celles de la commande *Coller*. Lorsque l'objet déplacé est affiché dans le format propre à l'application, utilisez les commandes d'achèvement de transfert telles que *Déplacer ici* et *Copier ici* dans le menu autonome. Lorsque l'objet déplacé est affiché sous un autre format, incluez le type de l'objet dans le nom des commandes. Vous pouvez également présenter d'autres commandes d'achèvement de transfert qui communiquent le contexte de la destination. Par exemple, une imprimante pourrait afficher la commande *Imprimer ici*. Dans le cas de commandes qui supportent seulement un aspect partiel ou une transformation d'un objet, utilisez des indicateurs plus descriptifs : *Copier les propriétés ici* ou *Transposez ici*, par exemple.

Les autres formes de la commande *Glisser-Déposer* prennent la forme générale suivante :

[*Nom commande*] [*Nom type* | *Nom objet*] **Ici** [**sous** *Nom type*]

Voici les commandes d'achèvement de transfert les plus répandues :

Commande	Fonction
Déplacer ici	Déplace la sélection à l'emplacement destination sous le format propre à l'application.
Copier ici	Crée une copie de l'objet sélectionné à l'emplacement destination sous le format propre à l'application.
Déplacer [<i>nom type</i>] ici Copier [<i>nom type</i>] ici	Déplace ou copie l'objet sélectionné en tant qu'objet OLE incorporé. L'objet est affiché sous la forme de son contenu et peut être activé directement à partir de la destination.
Déplacer [<i>nom type</i>] ici en tant qu'icône Copier [<i>nom type</i>] ici en tant qu'icône	Déplace ou copie l'objet sélectionné en tant qu'objet OLE incorporé et l'affiche sous la forme de son icône.

L'une des commandes du menu autonome doit être définie et affichée, par exemple la commande *Glisser-Déposer par défaut*, qui aura le même effet que *Glisser-Déposer* avec le bouton 1 de la souris.

Annuler une opération de transfert *Glisser-Déposer*

Lorsqu'un utilisateur glisse et dépose un objet sur lui-même, interprétez cette action comme l'annulation d'une manipulation directe de transfert. Également, annulez le transfert si l'utilisateur appuie sur la touche ÉCHAP pendant le déplacement.

Différencier le transfert de la sélection lors d'une action *Glisser*

Parce que *Déplacer* permet à la fois la sélection et le transfert des objets, il faut fournir à l'utilisateur une convention lui permettant de distinguer ces deux opérations. La convention à utiliser dépend du contexte courant dans lequel se situe l'objet ; il est également possible de fournir à l'utilisateur des poignées propres à la sélection ou au transfert. Les techniques les plus répandues se réfèrent à la position du pointeur au début de l'opération pour déterminer son type. Si le pointeur est situé sur une sélection existante, interprétez l'action comme étant une commande pour *Déplacer*. Si l'action débute à l'extérieur de la sélection, dans l'espace blanc de l'arrière-plan, interprétez l'action comme une opération de sélection.

Faire défiler lors d'un transfert avec une action Glisser

Lorsque l'utilisateur glisse et déplace un objet d'une zone défilante (une fenêtre, un panneau ou une zone de liste) vers une autre, certaines tâches exigent le transfert de l'objet à l'extérieur des limites de la zone en question. D'autres tâches pourraient exiger le transfert de l'objet dans une zone non affichée à l'écran. Dans ce dernier cas, il est pratique de faire défiler la zone de manière automatique (*autodéfilement*) lorsque l'utilisateur déplace l'objet jusqu'à ses limites. Vous pouvez implanter deux types de comportements en fonction de la vitesse de l'action *Déplacer*. Par exemple, si l'utilisateur déplace lentement l'objet aux limites de la zone, faites-la défiler ; si l'objet est déplacé rapidement, ne faites pas défiler la zone.

Pour implanter cette technique lors d'une opération *Déplacer*, échantillonnez la position du pointeur au début de l'action, chaque fois que la souris bouge, ainsi qu'avec une horloge incorporée à votre application (fréquence d'échantillonnage recommandée : toutes les 100 Ms). Si vous utilisez un soutien OLE à l'action *Glisser-Déposer*, vous n'avez pas besoin d'horloge incorporée. Emmagasinage les données dans un vecteur mémoire suffisamment grand pour contenir un minimum de trois échantillons simultanément, en remplaçant les anciens échantillons par de nouveaux. Ensuite, calculez la vitesse du pointeur en fonction de ses deux dernières positions.

Pour calculer la vitesse, additionnez les distances entre les points des deux échantillons et divisez le total par le temps écoulé entre les deux échantillons. La distance est la valeur absolue de la différence entre les positions en x et en y, ou $(\text{abs}(x1 - x2) + \text{abs}(y1 - y2))$. Multipliez cette valeur par 1024 et divisez-la par le temps écoulé entre les échantillons. La multiplication par 1024 prévient la perte de précision causée par la division de nombres entiers.

Note : La distance telle que présentée dans l'algorithme précédent n'est pas véritablement cartésienne. Cet algorithme utilise une approximation dans un but d'efficacité, plutôt que d'utiliser la racine carrée de la somme des carrés des distances latérales $((x1 - x2)^2 + (y1 - y2)^2)^{1/2}$, qui entraîne des coûts de calcul plus élevés.

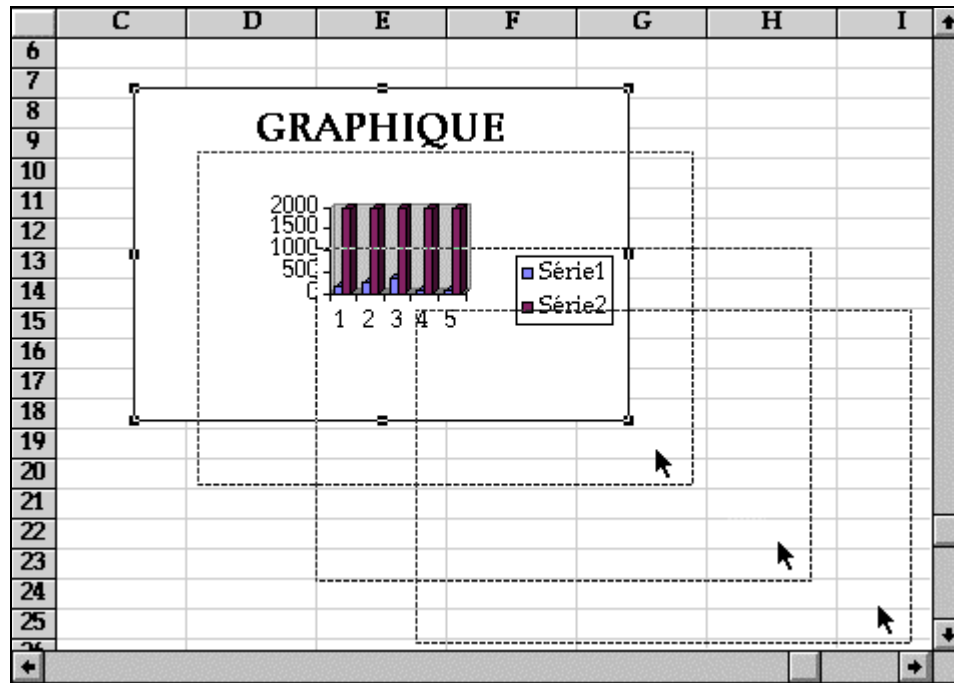
Également, définissez une zone sensible longeant les limites de la zone de défilement ainsi qu'un délai d'arrêt de défilement. Pour déterminer la largeur de la zone sensible, utilisez deux fois la largeur d'une barre de défilement verticale ou encore deux fois la hauteur d'une barre de défilement horizontale.

Pendant l'opération *Déplacer*, faites défiler la zone (défilante) si les conditions suivantes sont remplies :

- l'utilisateur déplace le pointeur à l'intérieur de la zone sensible ;
- la vitesse courante est en deçà d'un certain seuil ;
- la zone défilante peut défiler dans la direction associée à la zone sensible.

La vitesse-seuil recommandée est de 20 pixels par seconde. Ces conventions sont illustrées à la figure qui suit :

Figure 16 : Défilement automatique en fonction de la vitesse de déplacement du pointeur



La granularité de défilement dépend du type d'information et de la distance raisonnable de défilement. Par exemple, pour du texte, il faut habituellement faire défiler verticalement une ligne à la fois. L'approche généralement conseillée consiste à utiliser la même granularité que celle des flèches de barres de défilement.

Déterminez la fréquence de défilement que vous voulez implanter (ex. : quatre lignes par seconde). Après avoir vérifié la vitesse de défilement pour lancer l'*autodéfilement*, déclenchez l'horloge pour une période fixée (ex. : 100 Ms). Au terme de ce délai, calculez le temps écoulé depuis le dernier défilement. Si le temps écoulé est plus long que votre période de défilement, faites défiler d'une unité. Sinon, remettez votre horloge à zéro et vérifiez à nouveau lorsque le délai est écoulé.

3.2.3 Retour d'information de transfert

Parce que le transfert des objets est l'une des sources d'erreurs les plus répandues chez les utilisateurs, le retour d'information est un élément important de la conception de l'interaction. Un retour d'information insuffisant ou incohérent peut semer la confusion chez l'utilisateur.

Transfert par la méthode des commandes

Pour un transfert par la méthode des commandes, retirez visuellement l'objet sélectionné lorsque l'utilisateur choisit la commande *Couper*. Si, dans certaines circonstances, le retrait de l'objet est impraticable, vous pouvez alors afficher l'objet sélectionné sous une nouvelle apparence afin d'informer l'utilisateur que la commande *Couper* a été effectuée, mais que le transfert est toujours en attente de traitement. Par exemple, le système affiche les icônes sous leur forme ombrée ou de contour pour indiquer cet état d'attente de traitement. L'utilisateur s'attend habituellement à ce que la commande *Couper* retire la sélection : il faut donc évaluer soigneusement l'impact que cette solution alternative de retour

d'information aura sur la cohérence de l'interaction. Il vous faudra également restituer l'état visuel de l'objet si l'utilisateur choisit les commandes *Couper* ou *Copier* pour un autre objet avant d'avoir choisi la commande *Coller*, ce qui annule la commande *Couper* en attente de traitement.

La commande *Copier* n'exige aucun retour d'information particulier. La commande *Coller* a pour seul retour d'information l'insertion de l'objet déplacé. Toutefois, si l'objet affiché n'a pas été retiré lors de l'exécution de la commande *Couper* et que vous avez utilisé une forme alternative de représentation, vous devez la retirer maintenant.

Transferts par manipulation directe

Lors d'une opération de manipulation directe, présentez un retour d'information visuel pour l'objet, le pointeur ainsi que la destination. Plus précisément :

- Affichez l'objet sous son apparence d'objet sélectionné lorsque la vue dans laquelle il se trouve est le centre d'attention. Afin d'indiquer que l'objet est en état de transfert, vous pouvez ajouter de nouvelles caractéristiques à l'apparence de l'objet. Par exemple, pour une opération *Déplacer*, vous pouvez employer la forme ombrée (ou de contour) de l'objet, comme le fait le système pour indiquer qu'un objet a été coupé. Lorsque cela est possible, changez l'apparence visuelle de l'objet déplacé en fonction de l'opération de complétion de transfert par défaut permise par la destination où se trouve le pointeur. Conservez la représentation de l'objet à son emplacement original tant que l'utilisateur n'a pas complété l'opération de transfert. Cela permet non seulement de donner un indice visuel quant à la nature de l'opération de transfert mais également de fournir un point de référence visuel très utile.
- Affichez une forme de représentation de l'objet qui se déplacera avec le pointeur. Utilisez une représentation qui donnera des indications à l'utilisateur sur la manière dont l'objet apparaîtra à l'emplacement destination mais qui ne nuira pas à la visualisation de l'emplacement d'insertion de l'objet. Par exemple, lors du transfert d'un objet dans un contexte textuel, le point d'insertion ne doit pas être caché par l'opération *Déplacer*. Les représentations translucides ou de contour sont particulièrement appropriées car elles permettent de voir l'emplacement d'insertion tout en donnant de l'information sur la taille, la position et la nature de l'objet en transfert (voir la figure suivante).

Figure 17 : Représentations de contour et translucides pour les opérations de transfert

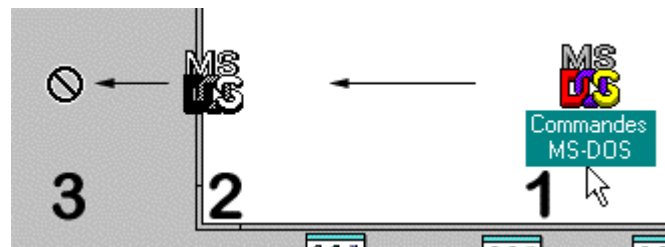


L'emplacement source de l'objet fournit l'apparence initiale de ce dernier, mais les destinations peuvent la modifier. La présentation de l'objet à l'écran doit donner des indications quant à la manière dont

l'objet sera intégré par sa destination. Si un objet est intégré en tant qu'icône, présentez-le sous sa forme iconique. Si l'objet est intégré dans le format propre à l'application, sa présentation devrait refléter cet état de fait. Par exemple, si un tableau déplacé dans un autre document est intégré en tant que tableau, la représentation de transfert pourrait être un tableau en contour ou translucide. Par contre, si le tableau est converti en texte, représentez-le sous forme de texte : vous pourriez afficher les premiers mots du tableau en représentation translucide.

- Affichez le pointeur d'insertion approprié en fonction du contexte de la destination. Par exemple, dans un contexte d'édition de texte, lorsqu'un objet doit s'insérer entre deux caractères, affichez le pointeur d'édition (parfois appelé pointeur poutre en I – *I beam pointer*)
- Présentez un retour d'information visuel pour indiquer la réceptivité d'éventuelles destinations. Vous pouvez employer le surlignage des sélections ou encore animer ou afficher une représentation de l'objet de transfert à l'emplacement destination. Vous pouvez également indiquer que la destination refuse l'objet déplacé en utilisant le pointeur *Ne peut déplacer l'objet à cet emplacement*, tel qu'illustré à la figure qui suit. Ce pointeur apparaît seulement lorsque le pointeur est situé au-dessus de la destination interdite.

Figure 18 : Pointeur *Ne peut déplacer l'objet à cet emplacement*



3.2.4 Commandes de transfert spécialisées

Dans certains contextes, une forme spécifique de transfert peut être si répandue qu'il devient approprié d'ajouter une commande de transfert spécialisée. Par exemple, si la copie d'objets est une opération fréquente, vous pouvez inclure une commande *Dupliquer*. La liste suivante présente les commandes de transfert spécialisées les plus répandues.

Commande	Fonction
Supprimer	Retire un objet de son contenant. Si l'objet en question est un fichier, il est détruit (<i>Delete</i>).
Détruire	Élimine le contenu d'un contenant (<i>Clear</i>).
Dupliquer	Copie les objets sélectionnés.
Imprimer	Imprime les objets sélectionnés sur l'imprimante par défaut.
Envoyer vers	Affiche une liste de destinations de transfert possibles et déplace l'objet vers la destination sélectionnée par l'utilisateur.

Note : Les commandes *Supprimer* et *Détruire* sont souvent employées de manière identique. Toutefois, elles se différencient par l'objet sur lequel elles s'appliquent : la commande *Supprimer* s'applique à un objet et la commande *Détruire* au contenu d'un contenant.

3.2.5 Raccourci au clavier pour les opérations de transfert

La liste suivante présente les techniques de raccourci au clavier pour les opérations de transfert :

Raccourci	Opération
CTRL+X	Effectue une commande <i>Couper</i> .
CTRL+C	Effectue une commande <i>Copier</i> .
CTRL+V	Effectue une commande <i>Coller</i> .
CTRL+Glisser	Passe de l'opération de transfert par manipulation directe par défaut à une opération <i>Copier</i> , à condition que la destination permette cette opération. La modification peut s'effectuer avec n'importe lequel des boutons de la souris.
ESC	Annule une opération de transfert <i>Glisser-Déposer</i> .

L'usage de ces raccourcis au clavier étant très répandu, assurez-vous de ne pas les réassigner à d'autres commandes.

3.3 OPÉRATIONS DE CRÉATION

La création de nouveaux objets est une action de l'utilisateur répandue dans les interfaces. Bien que les applications fournissent généralement le contexte de création des objets, vous devez éviter de considérer l'interface d'une application comme unique moyen de création de nouveaux objets. La création est généralement basée sur des spécifications ou des objets prédéfinis et peut être implantée de nombreuses manières.

3.3.1 Commande *Copier*

Le paradigme fondamental de la création de nouveaux objets consiste à faire des copies d'objets existants. Les objets copiés peuvent être modifiés et servent de prototype pour la création d'autres nouveaux objets. Les conventions du modèle de transfert définissent les techniques d'interaction de base pour la copie des objets. Les commandes *Couper* et *Coller* ainsi que la manipulation *Glisser-Déposer* permettent cette interaction.

3.3.2 Commande *Nouveau*

La commande *Nouveau* facilite la création de nouveaux objets. Elle s'applique à un objet spécifique, créant automatiquement une nouvelle instance de ce dernier. Cette commande est différente des commandes *Couper* et *Coller* en ce qu'elle permet de produire de nouveaux objets en une commande unique.

3.3.3 Commande *Insérer*

La commande *Insérer* fonctionne de manière similaire à la commande *Nouveau*, sauf qu'elle s'applique à un contenant pour créer un nouvel objet, généralement d'un type spécifique, à l'intérieur de ce même contenant. En plus d'insérer des données dans le format propre à l'application, vous pouvez utiliser cette commande pour insérer des objets de types différents. En implantant OLE, vous permettez la création d'un vaste ensemble d'objets. De plus, les objets créés dans votre application pourront être insérés dans des fichiers créés par d'autres applications OLE.

3.3.4 Utilisation des contrôles

Vous pouvez utiliser des contrôles pour permettre la création de nouveaux objets. Par exemple, dans les applications de dessin, il est courant d'employer des boutons pour la sélection de modes et d'outils lors de la création de nouveaux objets, tels que des formes spécifiques. Les boutons peuvent également être utilisés pour insérer des objets OLE.

4. FENÊTRES

L'environnement Windows offre le cadre fondamental par lequel l'utilisateur perçoit les données et interagit avec elles. Cette section décrit les types de fenêtres les plus répandus dans l'environnement Windows 3.1 et présente des lignes directrices de conception et d'apparence de ces fenêtres. Les versions ultérieures de Windows, telles que Windows NT et Windows 95, présentent des apparences et des comportements différents : l'information traitée dans cette section est propre à l'environnement Window 3.1.

4.1 TYPES DE FENÊTRES

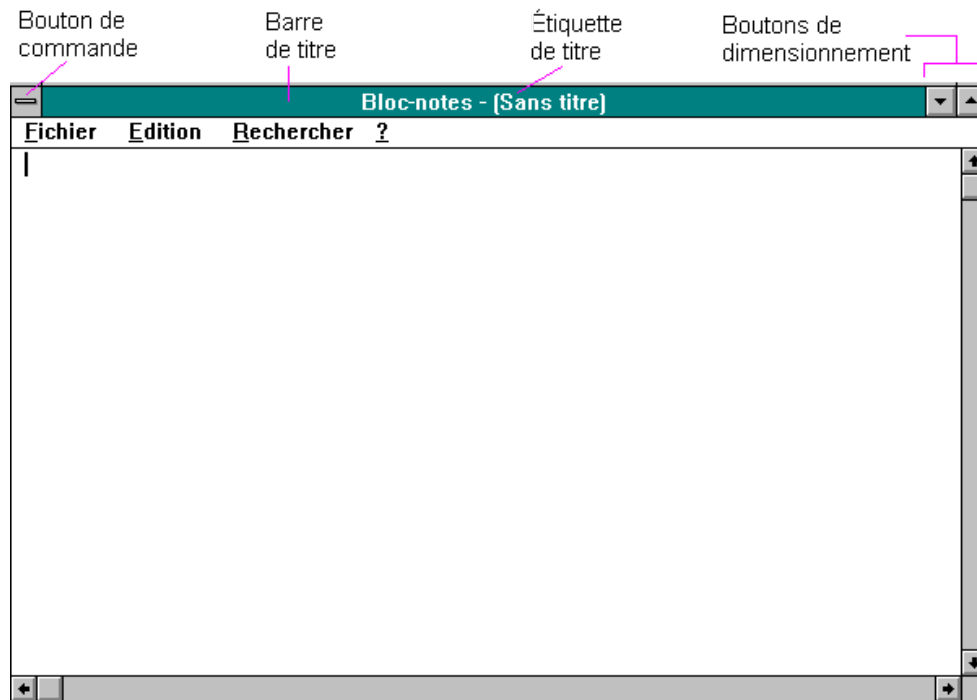
Les fenêtres sont classées en fonction de l'utilisation que l'on en fait. L'interaction avec les objets implique typiquement une fenêtre principale, où se déroulent les activités courantes d'affichage et d'édition. Plusieurs fenêtres secondaires peuvent être ajoutées pour permettre aux utilisateurs de spécifier certains paramètres ou options, ou pour fournir des détails particuliers à propos des objets ou des actions de la fenêtre principale.

4.2 COMPOSANTES DE LA FENÊTRE PRINCIPALE

Une fenêtre principale typique est délimitée par un cadre (ou bordure) et comprend une barre de titre qui identifie le contenu affiché dans la fenêtre. Si le contenu à afficher dépasse la taille de la fenêtre, des barres de défilement sont utilisées. Ces dernières peuvent être situées le long de la bordure de droite de la fenêtre principale, de la bordure du bas, ou encore aux deux endroits à la fois. La fenêtre peut également inclure d'autres composantes, telles que des barres de menu, des barres d'outils, ou encore des barres d'état.

La figure suivante montre certaines composantes d'une fenêtre principale :

Figure 19 : Fenêtre principale



Une fenêtre redimensionnable possède une bordure particulière. Lorsque l'utilisateur déplace le pointeur de la souris au-dessus de cette bordure, le pointeur change d'apparence pour indiquer qu'il est possible de redimensionner la fenêtre par manipulation directe. Si la fenêtre ne peut être redimensionnée, la bordure coïncide avec les limites de la fenêtre.

Au sommet de la fenêtre, à l'intérieur de la bordure, se trouve la barre de titre : elle identifie le contenu affiché dans la fenêtre. Le texte de la barre de titre doit correspondre au texte de l'icône du gestionnaire de fichier qui correspond à l'objet affiché. La barre de titre sert également de point de contrôle pour accéder à la fenêtre et la déplacer.

La barre de titre comporte, à son extrémité gauche, un bouton de commande qui permet d'accéder aux commandes associées à cette fenêtre, telles que *Restauration*, *Déplacement*, *Dimension*, *Réduction*, *Agrandissement*, *Fermeture* et *Basculer vers*. Double-Cliquer sur ce bouton de commande permet de fermer la fenêtre principale. À l'extrémité droite de la fenêtre, on trouve deux icônes de dimensionnement permettant de réduire ou d'agrandir la taille de la fenêtre.

4.3 OPÉRATIONS WINDOWS DE BASE

Les opérations de base de l'environnement Windows sur les fenêtres sont les suivantes : l'activation et la désactivation, l'ouverture et la fermeture, le déplacement, le redimensionnement, le défilement et le fractionnement. Les outils de développement les plus répandus gèrent efficacement toutes les opérations de base de l'environnement fenêtre.

4.3.1 Activation et désactivation

Bien que le système permette l’affichage simultané de plusieurs fenêtres, l’utilisateur ne travaille généralement que dans une seule fenêtre, appelée la fenêtre active. Des opérations peuvent être traitées dans les fenêtres inactives mais seule la fenêtre active peut recevoir les données de l’utilisateur. La fenêtre active se distingue par la couleur de sa barre de titre, différente de celle des fenêtres inactives.

L’utilisateur active une fenêtre en basculant (*switching*) à l’intérieur, ce qui rend les autres fenêtres inactives. L’utilisateur peut basculer dans une fenêtre en cliquant dessus avec la souris, en naviguant à l’aide des raccourcis au clavier ou en choisissant son icône si la fenêtre est réduite.

4.3.2 Ouverture et fermeture des fenêtres

L’ouverture d’une fenêtre rend cette dernière active et la fait placer sur le dessus du bureau, en fonction de son ordre en Z. Lorsque vous ouvrez une fenêtre principale pour la première fois, donnez-lui une taille et un emplacement raisonnables en fonction de l’objet ou de l’application.

Parce que la résolution et l’orientation (portrait ou paysage) des applications sont diverses, votre application ne devrait pas fixer la taille de son affichage ; elle devrait plutôt s’adapter à la taille et à la forme adoptées par le système d’exploitation. Si vous utilisez les interfaces système standard, le système placera automatiquement les fenêtres de votre application en fonction de l’affichage en cours.

L’utilisateur ferme la fenêtre en double-cliquant sur le bouton de gauche de la barre de titre, en utilisant la commande *Fermer* du menu de la fenêtre ou en choisissant la commande *Quitter* du menu *Fichier* de l’application. Lorsque l’utilisateur souhaite fermer la fenêtre, si votre application n’effectue pas l’enregistrement automatique des modifications et des traitements en cours, affichez un message permettant à l’utilisateur d’effectuer (ou pas) l’enregistrement des modifications ou d’annuler la commande *Fermer*.

4.3.3 Déplacement des fenêtres

L’utilisateur peut déplacer une fenêtre soit en faisant glisser sa barre de titre à l’aide de la souris, soit en choisissant la commande *Déplacer* du menu de la fenêtre. Après avoir choisi cette commande, l’utilisateur peut déplacer la fenêtre à l’aide des touches fléchées puis en appuyant sur la touche ENTRÉE pour terminer l’opération.

Ne permettez pas à l’utilisateur de déplacer la fenêtre à un emplacement où il ne lui sera plus possible d’y avoir accès par la suite.

Le déplacement de la fenêtre peut couper ou révéler certaines informations affichées. Lorsque l’utilisateur déplace une fenêtre, ne pas modifier le contenu affiché à l’intérieur.

4.3.4 Redimensionnement des fenêtres

Autorisez le redimensionnement des fenêtres, à moins que la taille n’en soit fixée, comme pour la calculatrice Windows, par exemple. L’utilisateur peut redimensionner la fenêtre en faisant glisser des bordures avec la souris, ou encore à l’aide de la commande *Dimension* du menu de la fenêtre.

Lorsque l’utilisateur réduit la taille de la fenêtre, vous devez déterminer comment disposer l’information dans cette fenêtre. L’approche la plus courante consiste à couper l’information, mais vous pouvez égale-

ment choisir de boucler (dans le cas d'un contenu texte) ou de redimensionner les textes. Servez-vous du contexte et du type d'information à afficher pour faire votre choix.

La commande *Réduction* (*Minimize*) réduit la fenêtre à sa taille minimale. Pour une fenêtre principale, cela signifie retirer la fenêtre du bureau et l'afficher sous sa forme iconique. La commande *Agrandissement* (*Maximize*) augmente la taille de la fenêtre de manière à ce qu'elle occupe tout l'espace du bureau. La commande *Restauration* (*Restore*) permet de retourner à la taille précédente de la fenêtre après une opération *Réduction* ou *Agrandissement*.

4.3.5 Défilement des fenêtres

Lorsque l'information affichée dans une fenêtre dépasse la taille de la fenêtre, vous devez permettre le défilement. Cette technique permet à l'utilisateur de visualiser des portions de l'objet qui ne sont pas actuellement visibles dans la fenêtre. Le défilement s'effectue à l'aide des barres de défilement, situées le long des bordures droite et inférieure de la fenêtre, selon le besoin.

La barre de défilement est composée de deux flèches, à chaque extrémité de la barre, et d'un curseur de défilement, aussi appelé bouton de défilement, ascenseur ou bouton-glisseur. Le bouton de défilement sert à indiquer, de manière continue, dans quelle portion du document on se trouve (au début, au milieu, à la fin...). Les flèches de défilement servent à se déplacer dans le document. La taille du bouton de défilement peut varier afin d'indiquer la proportion de la partie de document affichée par rapport au document entier.

4.3.6 Fractionnement des fenêtres

Une fenêtre peut être fractionnée en deux panneaux d'affichage différents ou davantage afin d'offrir plus d'aires de visualisation. Cette technique est très utile pour visualiser simultanément différentes parties d'un même document, par exemple. Elle peut également servir à afficher des vues différentes de la même information.

Pour permettre le fractionnement d'une fenêtre, ajoutez la commande *Fractionner* au menu *Fenêtre* de la barre de menu ou placez une boîte de fractionnement sur la barre de défilement. La boîte de fractionnement est généralement située au sommet de la barre de défilement verticale ou encore à la gauche de la barre de défilement horizontale. L'utilisateur peut fractionner la fenêtre en glissant le pointeur de la souris de la boîte de fractionnement à la position désirée ; double-cliquer sur la boîte de fractionnement peut également être un raccourci pour fractionner la fenêtre en son centre. Lorsque l'utilisateur déplace le pointeur de fractionnement, faites apparaître une ligne de fractionnement qui lui indiquera où la coupure aura lieu.

L'utilisateur peut enlever la ligne de fractionnement de deux manières : en choisissant la commande *Défaire fractionnement* ou encore en double-cliquant sur la boîte de fractionnement.

5. FENÊTRES SECONDAIRES

La plupart des fenêtres principales ont besoin d'un ensemble de fenêtres secondaires servant au soutien des activités de l'utilisateur dans la fenêtre principale. Cette section couvre les usages les plus répandus des fenêtres secondaires, tels que les feuilles de propriétés, les boîtes de dialogue, les palettes et les boîtes de messages.

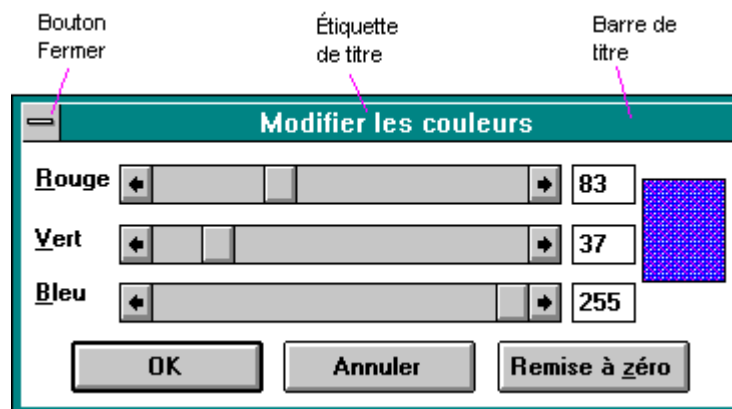
5.1 CARACTÉRISTIQUES DES FENÊTRES SECONDAIRES

Bien que les fenêtres secondaires possèdent quelques caractéristiques en commun avec les fenêtres principales, elles en diffèrent par leur comportement et par l'usage que l'on en fait. Par exemple, les fenêtres secondaires ne sont pas affichées dans la fenêtre *Liste de tâche* indiquant quelles sont les applications en cours. Les fenêtres secondaires demandent ou affichent des informations supplémentaires qui sont, le plus souvent, associées à des objets qui apparaissent dans une fenêtre principale.

5.1.1 Apparence et comportement

Une fenêtre secondaire typique, telle qu'illustrée à la figure suivante, comprend une barre de titre et un cadre : l'utilisateur peut déplacer cette fenêtre en faisant glisser la barre de titre à l'aide de la souris. Toutefois, une fenêtre secondaire ne possède pas de boutons *Réduction* et *Agrandissement* car ces opérations de dimensionnement ne s'appliquent pas, en général, à ce type de fenêtre. Un bouton *Fermer* peut être inclus à l'extrême gauche de la barre de titre pour retirer la fenêtre de l'écran, soit en double-cliquant sur la boîte ou en choisissant la commande *Fermer* du menu associé. Le texte de la barre de titre est une étiquette qui décrit le but de la fenêtre : le contenu de cette étiquette dépend de l'utilisation de la fenêtre.

Figure 20 : Fenêtre secondaire



Vous pouvez indiquer des informations d'état dans une fenêtre secondaire mais évitez d'employer une barre d'état comme celle utilisée dans les fenêtres principales.

Comme pour les fenêtres principales, les fenêtres secondaires peuvent posséder leurs propres menus autonomes de commandes. L'utilisateur accède au menu autonome d'une fenêtre secondaire à l'aide des mêmes techniques d'interaction que pour les fenêtres principales.

Interaction avec les autres fenêtres

Les fenêtres secondaires qui sont affichées par l'entremise de commandes de la fenêtre principale dépendent de l'état de cette dernière. En pratique, cela signifie que si une fenêtre principale est fermée ou minimisée, ses fenêtres secondaires sont également fermées ou cachées. Lorsque l'utilisateur rouvre ou restaure une fenêtre principale, restaurez les fenêtres secondaires à leurs anciens état et position. Toutefois, si l'ouverture d'une fenêtre secondaire est le résultat d'une action effectuée à l'extérieur d'une fenêtre principale, alors cette fenêtre secondaire est indépendante et est affichée au même titre que n'importe quelle fenêtre principale. Lorsque l'utilisateur ouvre une fenêtre secondaire ou y bascule, cette dernière est activée, comme toute autre fenêtre. À l'aide de la souris ou du stylet, l'utilisateur active une fenêtre secondaire de la même manière qu'une fenêtre principale. À l'aide du clavier, la combinaison ALT+F6 permet de basculer d'une fenêtre secondaire à la fenêtre principale ou encore à d'autres fenêtres secondaires liées à la même fenêtre principale. Pour permettre ce type de bascule, la fenêtre secondaire ne doit pas posséder plus d'un mode.

Lorsque l'utilisateur active une fenêtre principale, l'amenant ainsi à l'avant-plan des autres fenêtres selon un ordre en Z, toutes les fenêtres secondaires qui en dépendent (ayant été préalablement ouvertes) sont également affichées, tout en conservant leur position respective. De même, l'activation d'une fenêtre secondaire dépendante amène la fenêtre principale associée ainsi que les autres fenêtres secondaires à l'avant-plan.

Une fenêtre secondaire dépendante apparaît toujours au-dessus de la fenêtre principale associée ; s'il y a d'autres fenêtres secondaires (*pairs*) associées à la fenêtre principale, elles sont alors toutes disposées en couches successives par-dessus la fenêtre principale. Si un pair est activé, la première fenêtre secondaire activée demeure au-dessus de la fenêtre principale, mais au-dessous du pair nouvellement activé.

Vous pouvez concevoir une fenêtre secondaire de manière à ce qu'elle paraisse toujours au-dessus de ses pairs. En temps normal, vous ne devriez employer cette technique que pour les palettes ; même dans ce cas, permettez à l'utilisateur de reconfigurer cette caractéristique en ajoutant la commande *Toujours visible* aux commandes propres à cette fenêtre. Si vous supportez cette technique pour plusieurs fenêtres secondaires, la gestion des fenêtres s'effectue alors selon l'ordre en Z pour l'ensemble de fenêtres dont elles font partie.

Évitez d'afficher une fenêtre secondaire ayant le comportement *Toujours visible* par-dessus les fenêtres d'une autre application activée par l'utilisateur, à moins que cette fenêtre soit également utile pour la seconde application.

Lorsque l'utilisateur choisit une commande ouvrant une fenêtre secondaire, utilisez le contexte de l'opération afin de déterminer la manière de présenter l'information dans cette fenêtre. Par exemple, pour une feuille de propriétés, réglez les valeurs des propriétés de façon à représenter la sélection.

En général, affichez la fenêtre dans l'état où l'utilisateur l'avait laissée. Par exemple, une boîte de dialogue *Ouvrir* devrait conserver le répertoire d'accès courant entre chacune des ouvertures de la boîte de dialogue. De même, si vous employez des onglets pour naviguer parmi les informations d'une fenêtre secondaire, affichez l'onglet que consultait l'utilisateur lorsqu'il a fermé la fenêtre. Cette technique faci-

lite la répétition d'une opération associée à cette fenêtre pour l'utilisateur et confère davantage de stabilité à l'interface.

Toutefois, si une tâche ou une commande implique ou exige que l'utilisateur commence un traitement dans une séquence ou un état particulier, tel qu'une fenêtre *Assistant*, présentez plutôt la fenêtre secondaire selon un format cohérent ou encore un format fixe. Par exemple, l'entrée d'un élément dans une base de données pourrait exiger que l'utilisateur entre les données selon une séquence donnée. Par conséquent, il serait sans doute approprié que la fenêtre d'entrée affiche d'abord le premier champ de saisie.

Déploiement d'une fenêtre secondaire

À l'exception des palettes, évitez de rendre possible le redimensionnement des fenêtres secondaires car leur but est de présenter une information prédéfinie de manière concise. Toutefois, vous pouvez utiliser un *bouton de déploiement* pour accroître la taille d'une fenêtre de manière à révéler progressivement des options additionnelles. Un bouton de déploiement est un bouton de commande avec une étiquette comprenant deux caractères « plus grand que » (>>). Lorsque l'utilisateur choisit ce bouton, la fenêtre secondaire s'accroît jusqu'à une taille prédéterminée. De manière facultative, vous pouvez utiliser le bouton de nouveau pour « replier » la fenêtre.

Fenêtres secondaires en cascade

Vous pouvez également offrir à l'utilisateur l'accès à des options additionnelles en incluant un bouton de commande qui ouvre une autre fenêtre secondaire. Si cette nouvelle fenêtre s'active de façon indépendante, fermez la fenêtre secondaire d'où elle provient et n'affichez que la nouvelle fenêtre. Toutefois, si l'objet de cette nouvelle fenêtre est d'obtenir une information nécessaire à la fenêtre d'où elle provient, la fenêtre originale devra alors demeurer affichée ; la nouvelle fenêtre apparaît alors par-dessus, légèrement décalée vers la droite et sous la fenêtre originale. Lorsque vous utilisez cette méthode, limitez le nombre de fenêtres secondaires à un seul niveau, afin d'éviter un engorgement de chaînes de fenêtres hiérarchiques en cascade.

5.1.2 Positionnement des fenêtres

Lors de la détermination de l'emplacement d'une fenêtre secondaire, de nombreux facteurs sont à considérer, tels que l'utilisation qui sera faite de la fenêtre, les dimensions globales de l'affichage, ainsi que le pourquoi de l'affichage de cette fenêtre secondaire. En général, placez la fenêtre secondaire à son dernier emplacement connu. Si l'utilisateur n'a pas encore établi d'emplacement précis pour cette fenêtre, placez-la à un endroit qui permettra à l'utilisateur de naviguer aisément et de déployer entièrement la fenêtre. Si aucune de ces lignes directrices ne s'applique, centrez horizontalement la fenêtre secondaire à l'intérieur de la fenêtre principale, sous la barre de titre, sous le menu ou sous toute autre barre d'outils incorporée.

5.1.3 Fenêtre modale vs non modale

Une fenêtre secondaire peut être conçue avec ou sans mode. Une fenêtre *non modale* permet à l'utilisateur d'interagir avec la fenêtre secondaire comme avec la fenêtre primaire, tout comme l'utilisateur peut basculer entre les fenêtres principales de différentes applications. Elles sont également bien adaptées aux situations où l'utilisateur veut répéter une action : par exemple, trouver les occurrences d'un mot ou encore formater les propriétés d'un texte.

Une fenêtre secondaire *modale* exige que l'utilisateur complète l'interaction à l'intérieur de la fenêtre secondaire et ferme cette dernière avant de continuer toute autre interaction à l'extérieur de cette fenêtre.

Une fenêtre secondaire peut être modale par rapport à sa fenêtre principale ou encore par rapport au système. Dans ce dernier cas, l'utilisateur doit répondre et fermer la fenêtre avant d'interagir avec toute autre fenêtre ou application.

Parce que les fenêtres secondaires modales restreignent les choix de l'utilisateur, employez-les avec modération. Limitez leur utilisation aux situations où de l'information additionnelle est requise pour compléter une commande ou lorsqu'il est important d'empêcher toute interaction avant la réalisation d'une condition. Évitez d'employer des fenêtres secondaires modales, à moins que votre application n'interagisse directement avec le système ; même dans ce cas, ne les utilisez que dans des situations graves. Par exemple, lorsqu'une erreur système fatale ou une condition irrécupérable est imminente.

5.1.4 Boutons par défaut

Lors de la définition d'une fenêtre secondaire, vous pouvez attribuer la touche ENTRÉE (*ENTER*) à l'activation d'un bouton de commande en particulier, appelé *bouton par défaut*. Le système différencie le bouton par défaut des autres boutons de la fenêtre par une bordure épaisse entourant le bouton.

Définissez l'action du bouton par défaut comme l'action la plus probable, telle qu'une confirmation ou une action qui applique les transformations effectuées dans la fenêtre secondaire. Ne créez pas de boutons par défaut avec des commandes à effet destructif ou irréversible. Par exemple, dans une fenêtre de recherche et de substitution de texte, il ne faut pas employer le bouton *Effacer tout* comme bouton par défaut.

Vous pouvez changer le bouton par défaut pendant que l'utilisateur interagit avec la fenêtre. Par exemple, si l'utilisateur navigue vers un bouton de commande qui n'est pas celui par défaut, le nouveau bouton devient temporairement le bouton par défaut. Dans un tel cas, le nouveau bouton par défaut prend l'apparence des boutons par défaut (bordure épaisse) et l'ancien bouton par défaut reprend une apparence normale. De même, si l'utilisateur déplace le point d'attention vers un autre contrôle de la fenêtre qui n'est pas un bouton de commande, le bouton par défaut d'origine reprend son rôle.

L'attribution de boutons de commande par défaut est une convention répandue. Toutefois, lorsqu'il n'y a aucun bouton par défaut approprié ou encore qu'un autre contrôle requiert l'utilisation de la touche ENTRÉE (ex. : entrer une nouvelle ligne dans un contrôle de texte multiligne), n'attribuez pas de bouton par défaut pour cette fenêtre. Si un contrôle en particulier est le centre d'attention et requiert l'utilisation de la touche ENTRÉE, vous pouvez désactiver temporairement le bouton par défaut. Ensuite, lorsque l'utilisateur déplace le point d'attention hors de ce contrôle, vous pouvez restaurer le bouton par défaut.

De manière facultative, vous pouvez utiliser le double-clicquage sur un contrôle de sélection unique (un bouton à option ou une liste à sélection simple) comme technique de raccourci pour régler ou choisir une option et exécuter la commande par défaut de la fenêtre secondaire.

5.1.5 Navigation dans les fenêtres secondaires

Avec la souris ou le stylet, la navigation vers un champ ou un contrôle particulier implique le fait que l'utilisateur pointe sur le champ (ou le contrôle) en question et clique dessus. Pour les boutons de contrôle, cette action active le bouton. Par exemple, pour les cases à cocher, cette action « coche » ou « décoche » la case en question et pour un bouton de commande, elle exécute la commande associée au bouton.

L'interface clavier de navigation pour les fenêtres secondaires utilise les touches TAB et MAJ+TAB pour se déplacer entre les contrôles, respectivement le suivant et le précédent. Chaque contrôle possède une propriété qui détermine sa place dans l'ordre de navigation. Réglez cette propriété de telle manière que l'utilisateur puisse se déplacer dans la fenêtre selon les conventions usuelles de lecture. Placez les con-

trôles dans un ordre tel que l'utilisateur puisse progresser dans la fenêtre de manière logique, en se déplaçant parmi des groupes de contrôles associés. Les boutons de contrôle servant au traitement des transformations globales de la fenêtre sont généralement à la fin de la séquence de navigation.

Il n'est pas nécessaire de fournir un accès touche TAB à tous les contrôles situés dans la fenêtre. Lorsque vous utilisez du texte statique en guise d'étiquette, la destination de navigation devrait être le contrôle qui lui est associé et non pas le texte lui-même. De plus, aux fins de navigation, les contrôles combinés tels que les zones de liste modifiable (*combo box*), zones de liste déroulante modifiable (*drop-down combo box*) et les compteurs (*spin box*) sont considérés comme des contrôles simples. Puisque les boutons à option apparaissent généralement en groupe, utilisez la touche TAB pour déplacer le centre d'attention d'un groupe à l'autre mais non parmi les options d'un même groupe : utilisez les touches fléchées à cette fin. Pour un groupe de cases à cocher, utilisez la navigation par touche TAB car chaque option est indépendante des autres.

De manière facultative, lorsque l'interface ne requiert pas l'utilisation des touches fléchées, vous pouvez les utiliser en plus de la technique TAB pour permettre la navigation entre les contrôles. Par exemple, vous pouvez utiliser les touches FLÈCHE HAUT et FLÈCHE BAS pour naviguer entre des boîtes de texte à ligne simple ou encore parmi les éléments d'un groupe de cases à cocher. Utilisez toujours les touches fléchées pour naviguer parmi des choix de boutons à option ou à l'intérieur de zones de liste.

Vous pouvez également utiliser des touches d'accès rapide pour permettre la navigation parmi les contrôles dans une fenêtre secondaire. Cette technique permet à l'utilisateur d'accéder à un contrôle en appuyant et en maintenant enfoncée la touche ALT et la touche alphanumérique qui correspondent au caractère de touche d'accès rapide indiqué dans l'étiquette du contrôle.

Les touches alphanumériques sans modifications (i.e. sans association avec la touches ALT) permettent également la navigation si le contrôle qui détient le centre d'attention n'utilise pas ces touches comme forme d'entrée de données. Par exemple, si le centre d'attention est un contrôle de cases à cocher et que l'utilisateur appuie sur une touche alphanumérique, le centre d'attention se déplace vers le contrôle dont la touche d'accès rapide correspond à la touche enfoncée. Toutefois, si le centre d'attention est plutôt une zone de texte ou une zone de liste modifiable, l'utilisateur ne peut employer les touches alphanumériques pour naviguer sans les associer à la touche ALT : elles servent d'abord à l'entrée de données pour ce type de contrôle.

Les touches d'accès rapide permettent non seulement de naviguer vers le contrôle correspondant, elles ont également le même effet que de cliquer sur le contrôle avec la souris. Par exemple, appuyer sur la touche d'accès rapide d'un bouton de commande permet d'exécuter l'action associée à ce bouton. Afin d'assurer l'accès direct à tous les contrôles, choisissez des touches d'accès rapide uniques à l'intérieur d'une même fenêtre secondaire.

Vous pouvez également utiliser les touches d'accès rapide pour naviguer vers un contrôle, mais ensuite retourner le centre d'attention au contrôle d'où l'utilisateur provenait. Par exemple, lorsque l'utilisateur appuie sur la touche d'accès rapide d'une commande qui modifie le contenu d'une zone de liste, vous pouvez retourner le centre d'attention à la zone de liste après que la commande a été exécutée.

Lorsque les boutons de commande OK et *Annuler* sont les principales transformations pour une fenêtre secondaire, on ne leur attribue généralement pas de touches d'accès rapide. Dans ce cas, ce sont plutôt les touches ENTRÉE et ECHAP qui donnent accès à ces boutons de commande.

Appuyer sur la touche ENTRÉE équivaut à naviguer vers le bouton de commande par défaut, s'il y en a un, et à activer la commande qui lui est associée. S'il n'y a aucun bouton de commande par défaut, un contrôle peut alors utiliser la touche ENTRÉE à ses propres fins.

5.1.6 Validation des données

Dans une fenêtre secondaire, validez les données entrées par l'utilisateur dans un champ de données ou un contrôle le plus tôt possible après leur saisie. Idéalement, les données devraient être validées dès qu'elles sont entrées dans un champ en particulier. Si les données ne sont pas valides, vous pouvez les rejeter ou encore utiliser un retour d'information visuel ou auditif pour en informer l'utilisateur. Vous pouvez également afficher une boîte de message, surtout si l'utilisateur tente à de nombreuses reprises d'entrer des données non valides. De plus, vous pouvez réduire le retour d'information sur des données non valides en utilisant des contrôles qui limitent la sélection à un ensemble spécifique de choix (cases à cocher, cases à option, liste déroulante fixe, etc.) ou encore, prérégler le champ d'entrée avec une valeur par défaut raisonnable.

S'il n'est pas possible de valider les données au point d'entrée, envisagez la validation des données lorsque l'utilisateur quitte le contrôle. Si ce n'est pas possible, alors validez les données lorsque la transformation est exécutée, ou au moment où l'utilisateur tente de fermer la fenêtre. À ce moment, laissez la fenêtre ouverte et affichez un message ; après que l'utilisateur aura écarté le message, placez le centre d'attention sur le contrôle où se trouvent les données invalides.

5.2 BOÎTES DE DIALOGUE

Une boîte de dialogue permet un échange d'information (ou dialogue) entre l'utilisateur et l'application. Employez une boîte de dialogue pour obtenir de l'information additionnelle de la part de l'utilisateur : l'information nécessaire pour exécuter une commande ou une tâche particulière, par exemple.

Comme les boîtes de dialogue apparaissent généralement après l'activation d'un élément de menu ou d'un bouton de commande (incluant les menus autonomes ou en cascade), l'étiquette du titre de la boîte de dialogue doit être la même que l'étiquette de la commande associée. N'incluez pas d'ellipse (...) dans l'étiquette du titre, même si le nom de la commande en contenait une. De plus, évitez d'inclure le nom du menu auquel la commande appartient, à moins que cela ne s'avère nécessaire pour rendre le titre de la boîte de dialogue plus clair. Par exemple, pour la commande *Imprimer* du menu *Fichier*, le titre de la boîte de dialogue devrait être *Imprimer* et non pas *Imprimer...* ou *Fichier Imprimer*. Par contre, pour la commande *Objet...* du menu *Insérer*, vous pouvez utiliser l'expression *Insérer Objet* pour le titre de la boîte de dialogue.

5.2.1 Commandes des boîtes de dialogue

Tout comme les feuilles de propriétés, les boîtes de dialogue comportent généralement les boutons de commande OK et *Annuler*. Utilisez le bouton OK pour appliquer les valeurs de la boîte de dialogue puis fermez la fenêtre. Si l'utilisateur choisit la commande *Annuler*, les changements sont ignorés et la fenêtre est fermée, annulant ainsi l'opération choisie par l'utilisateur. Ces boutons de commande sont spécialement appropriés pour les boîtes de dialogue qui permettent à l'utilisateur de fixer les paramètres pour une commande donnée. Définissez généralement le bouton OK comme bouton par défaut lorsque la fenêtre est ouverte.

Vous pouvez également inclure d'autres boutons de commande dans les boîtes de dialogue, qui s'ajoutent ou qui remplacent les boutons OK et *Annuler*. Les étiquettes de vos boutons devraient indiquer clairement leur utilité, tout en demeurant aussi concises que possible. Les étiquettes longues et verbeuses rendent difficile l'interprétation du rôle exact d'un bouton. Suivre les conventions de conception des boutons de commande.

5.2.2 Disposition

Disposer les principaux boutons de commandes soit en les empilant le long de la bordure de droite en partant du haut, soit en les alignant au bas de la boîte de dialogue. Placez le bouton le plus important (généralement la commande par défaut) à la première place de la disposition choisie. Si vous utilisez les boutons OK et *Annuler*, regroupez-les ensemble. Vous pouvez utiliser d'autres formes de disposition pour des raisons valables, telles que des relations d'appariement naturel. Par exemple, il serait logique de placer des boutons étiquetés Nord, Sud, Est et Ouest selon une disposition en croix. De même, un bouton de commande qui modifie ou offre du soutien à d'autres contrôles pourrait être regroupé avec ou placé près de ces derniers. Toutefois, évitez de faire d'un tel bouton le bouton par défaut car l'utilisateur s'attend à trouver ce dernier à son emplacement habituel.

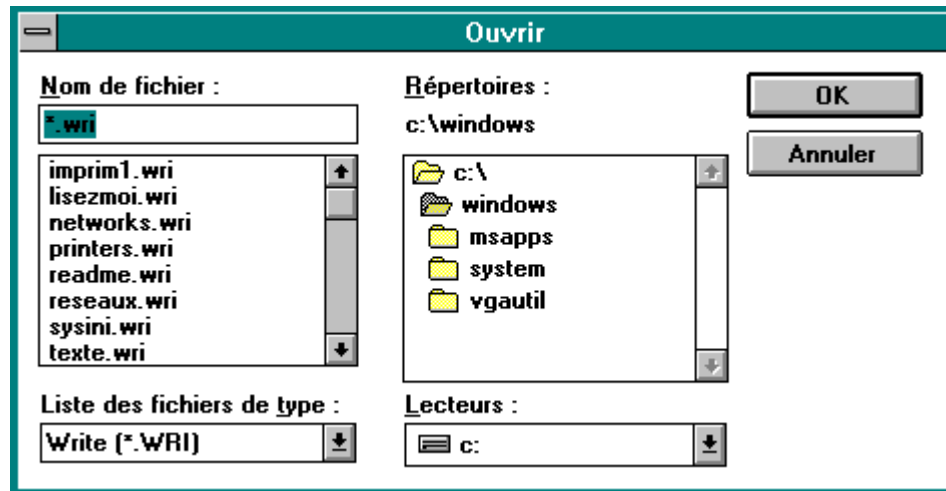
5.2.3 Interfaces de boîtes de dialogue répandues

Les outils de développement pour l'environnement Windows fournissent quelques interfaces déjà conçues pour plusieurs opérations répandues : utilisez-les lorsque la situation le permet. Elles vous permettent d'économiser du temps tout en assurant la cohérence de votre application. Si vous choisissez de concevoir vos propres interfaces, assurez-vous de maintenir la cohérence avec les fonctionnalités de base des interfaces standard ainsi qu'avec les lignes directrices qui guident leur usage. Par exemple, si vous concevez votre propre feuille de propriétés pour les caractères, assurez-vous que votre conception soit similaire, en apparence et en comportement, à la boîte de dialogue des caractères standard. La cohérence des styles visuels et opérationnels permettra aux utilisateurs de transférer leurs connaissances et leurs habiletés plus facilement.

Note : N'utilisez pas ces boîtes de dialogue pour créer ou mettre à jour votre application pour Microsoft Windows 95 ou des versions ultérieures; utilisez plutôt celles décrites dans [WIG95].

Boîte de dialogue *Ouvrir*

La boîte de dialogue *Ouvrir*, telle qu'illustrée à la figure suivante, permet à l'utilisateur de fureter dans le gestionnaire de fichier et comprend également des contrôles pour ouvrir un fichier spécifique. Utilisez cette boîte de dialogue pour ouvrir un fichier ou encore pour rechercher le nom d'un fichier, par exemple à l'aide de la commande *Ouvrir* du menu *Fichier*. Assurez-vous que le titre de la boîte de dialogue soit représentatif de sa commande.

Figure 21 : Boîte de dialogue *Ouvrir*

La boîte de dialogue fournie par le système gère automatiquement la validation des noms de fichiers et les transferts par manipulation directe (tels que *Glisser-Déposer*). La boîte de dialogue affiche les extensions de fichiers de types déposés (*registered*) seulement lorsque l'utilisateur choisit cette option d'affichage. Le filtre par défaut du champ *Liste des fichiers de type* est déterminé par votre application. Incluez les types de fichiers que votre application supporte dans la zone de liste déroulante *Liste des fichiers de type*. Pour chaque élément de la liste, utilisez de préférence les descriptions de types basées sur les noms de types déposés (*registered*). Par exemple, pour les fichiers de type texte, le descriptif de type devrait être *Documents Textes*. Vous pouvez également inclure un type *Tous* pour afficher tous les fichiers dans le répertoire courant, quel que soit leur type.

Pour ouvrir un fichier, l'utilisateur choisit le fichier dans la liste affichée dans la boîte de dialogue ou tape un nom de fichier dans le champ *Nom du fichier* ; il choisit ensuite la commande *Ouvrir* (ou *OK*). En guise de raccourci, il est également possible d'ouvrir le fichier en double-cliquant sur son icône. Choisir le bouton *Annuler* ferme la fenêtre sans ouvrir de fichier.

Les fichiers présentés dans la liste de la boîte de dialogue sont ceux inclus dans le répertoire courant qui correspondent au type de filtre fixé dans la liste déroulante *Liste des fichiers de type*. La zone de liste à arborescence *Répertoires* affiche le répertoire courant. L'affichage de la liste permet à l'utilisateur de visualiser la hiérarchie de répertoires et de naviguer dans l'arbre.

La boîte de dialogue *Ouvrir* ne gère que l'association entre un nom et un fichier. Il est de la responsabilité de votre application de s'assurer de la validité du format du fichier et d'en aviser l'utilisateur.

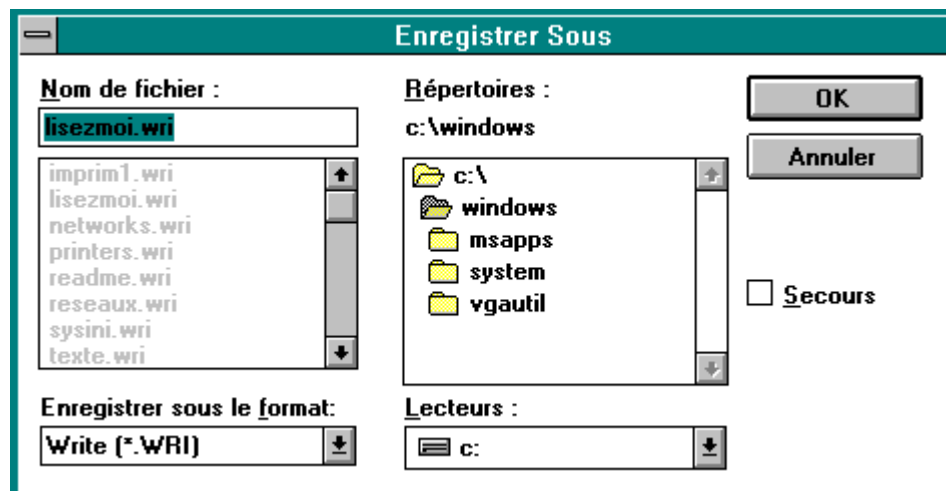
L'utilisateur peut changer le répertoire courant en choisissant un autre élément dans la liste *Répertoires*, en choisissant un contenant (tel qu'une chemise) de la liste de fichiers, ou en entrant au clavier un répertoire valide dans le champ *Nom du fichier* puis en activant le bouton *Ouvrir* (ou *OK*). L'activation du bouton *Annuler* ne devrait pas changer le répertoire courant. Conservez toujours le dernier répertoire utilisé entre les ouvertures successives de la boîte de dialogue. Si l'application permet l'ouverture de multiples fichiers, comme dans les conceptions MDI, fixez le répertoire courant à celui du dernier fichier ouvert, pas celui de la fenêtre enfant active.

Boîte de dialogue *Enregistrer sous*

La boîte de dialogue *Enregistrer sous*, illustrée à la figure suivante, sert à enregistrer un fichier sous un nom, un emplacement, un type ou un format donné. Généralement, les applications qui supportent la création de multiples fichiers par les utilisateurs fournissent cette commande. Toutefois, si votre application ne gère que des fichiers de données privés et effectue automatiquement leur mise à jour, cette boîte de dialogue peut ne pas être appropriée.

Affichez cette boîte de dialogue lorsque l'utilisateur choisit la commande *Enregistrer sous* ou une autre commande fichier ayant une fonction similaire, telle que la commande *Exporter*. Affichez également cette boîte de dialogue lorsque l'utilisateur choisit la commande *Enregistrer* et qu'il n'a pas encore donné ou confirmé un nom de fichier. Si vous utilisez cette boîte de dialogue pour d'autres tâches exigeant l'enregistrement de fichiers, définissez l'étiquette de la barre de titre pour refléter adéquatement cette commande.

Figure 22 : Boîte de dialogue *Enregistrer sous*



L'apparence et le comportement de la boîte de dialogue *Enregistrer sous* sont similaires à ceux de la boîte de dialogue *Ouvrir*, sauf que le champ de type (la zone de liste déroulante *Enregistrer sous le format*) définit le type par défaut du fichier enregistré ; il filtre également la liste des fichiers affichés dans la fenêtre.

Pour enregistrer un fichier, l'utilisateur active le bouton *Enregistrer* (ou OK) : le fichier est sauvegardé sous le nom apparaissant dans la zone de texte *Nom du fichier*. Bien que l'utilisateur puisse entrer au clavier un nom ou encore choisir un fichier dans la liste affichée, votre application peut indiquer automatiquement (« prérégler ») le nom courant du fichier. Si le fichier n'a pas encore été nommé, proposez un nom de fichier basé sur le type déposé du fichier (ex. : *Document Texte*).

La liste à arborescence *Répertoires* indique le contenant immédiat dans lequel sera déposé le fichier parmi les répertoires (ou chemises). L'utilisateur peut changer le répertoire à l'aide de ce contrôle et de la boîte de liste de fichiers. Si le fichier existe déjà, toujours l'enregistrer dans son emplacement original. Cela signifie que le répertoire courant pour la boîte de dialogue *Enregistrer sous* devrait toujours être réglé sur le dernier répertoire où le fichier a été enregistré. Si le fichier n'a jamais été enregistré, l'enregistrer dans le répertoire par défaut de votre application ou à l'emplacement spécifié par l'utilisateur (il peut entrer ce dernier au clavier ou à l'aide des contrôles de la boîte de dialogue).

Si l'utilisateur choisit le bouton *Annuler* dans la boîte de dialogue *Enregistrer sous*, ne pas enregistrer ni le nom du fichier ni les réglages de type ou de répertoire. Restaurer le répertoire original.

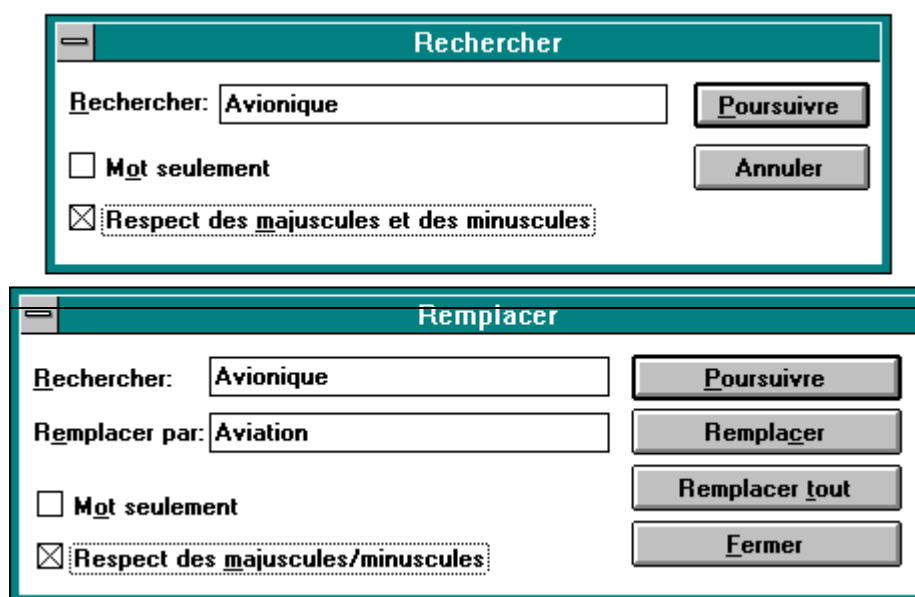
Lorsque l'utilisateur fournit un nom de fichier, la boîte de dialogue suit des conventions similaires à celles de la boîte de dialogue *Ouvrir*. Si l'utilisateur n'inclut pas d'extension, le système emploie l'extension spécifiée dans la zone *Enregistrer sous le format* ou l'extension par défaut de votre application. Si l'utilisateur inclut une extension, le système vérifie si l'extension correspond à l'extension par défaut de votre application ou à une extension déposée.

Note : Assurez-vous de conserver la date de création des fichiers que l'utilisateur ouvre et enregistre. Si votre application enregistre les fichiers en créant des fichiers temporaires qu'elle renomme avec les noms des fichiers d'origine ayant été préalablement détruits, assurez-vous de copier la date de création du fichier original. Certaines fonctionnalités du système de gestion de fichiers dépendent de l'identité du fichier original.

Boîtes de dialogue *Rechercher* et *Remplacer*

Les boîtes de dialogue *Rechercher* et *Remplacer* offrent des contrôles qui permettent de rechercher une chaîne de caractères spécifiée par l'utilisateur et, éventuellement, de la remplacer par une seconde chaîne de caractères spécifiée par l'utilisateur. Ces boîtes de dialogue sont illustrées à la figure suivante :

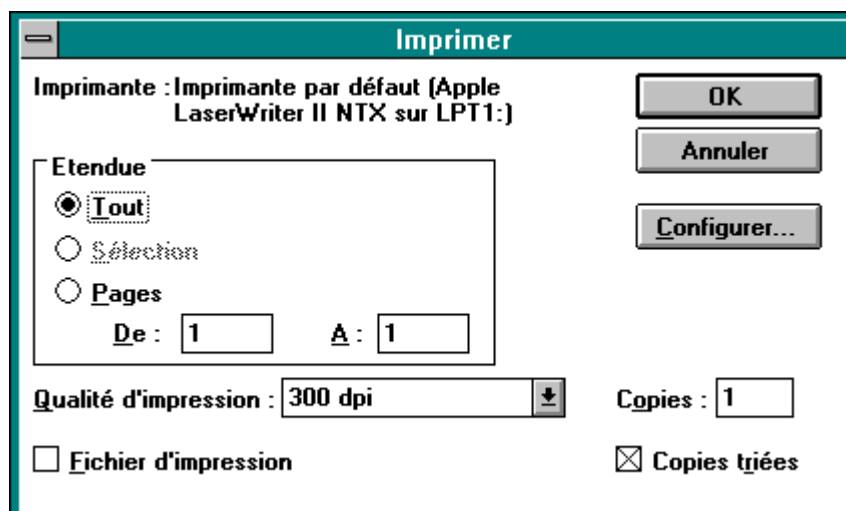
Figure 23 : Boîtes de dialogue *Rechercher* et *Remplacer*



Boîte de dialogue *Imprimer*

La boîte de dialogue *Imprimer*, illustrée à la figure suivante, permet à l'utilisateur de choisir ce qui sera imprimé, le nombre de copies ainsi que la séquence d'impression. Elle permet également à l'utilisateur de choisir une imprimante et fournit un bouton de commande qui offre un raccourci d'accès aux propriétés de l'imprimante.

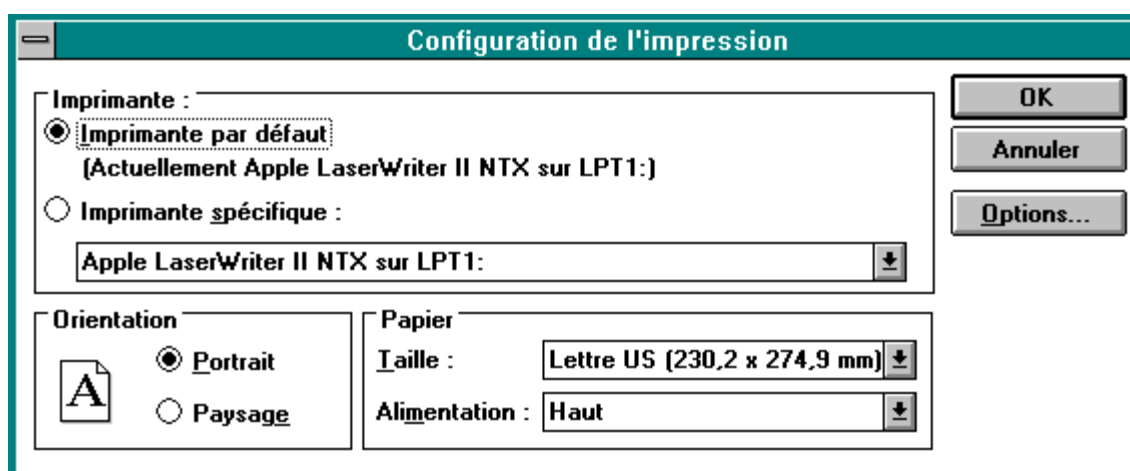
Figure 24 : Boîte de dialogue *Imprimer*



Boîte de dialogue *Configuration de l'impression*

La boîte de dialogue *Configuration de l'impression* (*Printer Setup*) affiche la liste des imprimantes disponibles et fournit les contrôles permettant de sélectionner une imprimante et de fixer l'orientation du papier, la taille, la source et autres propriétés de l'imprimante.

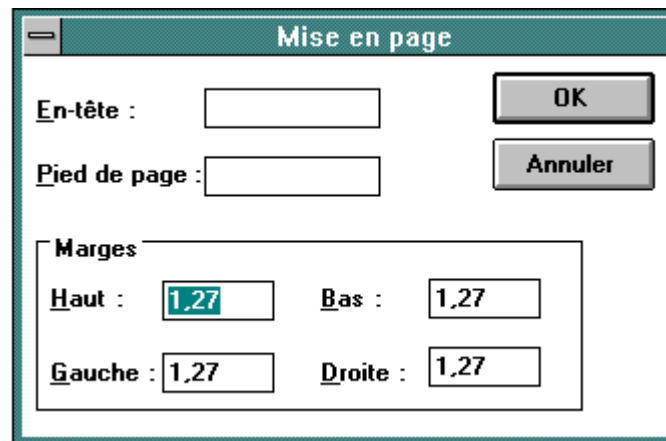
Figure 25 : Boîte de dialogue *Configuration de l'impression*



Boîte de dialogue *Mise en page*

La boîte de dialogue *Mise en page*, illustrée à la figure suivante, fournit les contrôles permettant de spécifier les propriétés et la disposition des éléments de la page.

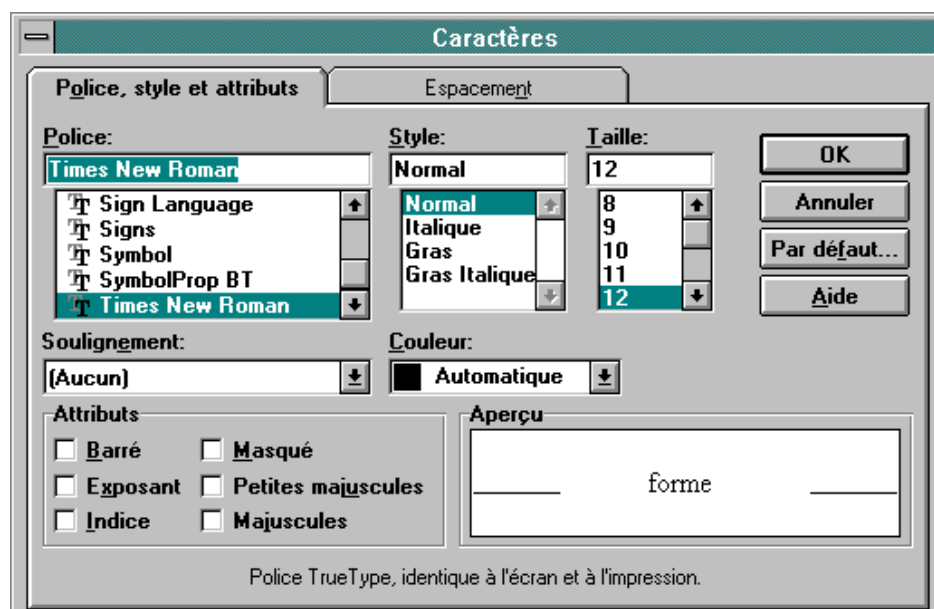
Figure 26 : Boîte de dialogue *Mise en page*



Boîte de dialogue *Caractères*

Cette boîte de dialogue affiche les polices de caractères disponibles ainsi que les tailles en point des polices installées sur le système. Votre application peut filtrer cette liste pour afficher seulement les polices autorisées par votre application. Utilisez la boîte de dialogue *Caractères* pour afficher ou fixer les propriétés des polices de caractères pour un texte sélectionné. La figure qui suit illustre la boîte de dialogue *Caractères*.

Figure 27 : Boîte de dialogue *Caractères*



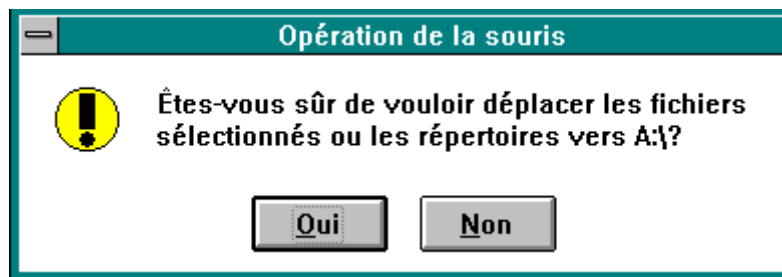
5.3 BOÎTES DE MESSAGE

Une boîte de message est une fenêtre secondaire qui affiche un message ou de l'information à propos d'une situation particulière.

5.3.1 Étiquette de la barre de titre

La boîte de message comporte une barre de titre dont l'étiquette permet d'identifier adéquatement le contexte dans le cadre duquel le message est affiché. Il est particulièrement important de fournir une étiquette appropriée dans l'environnement multitâche Windows car l'affichage des boîtes de message n'est pas toujours le résultat d'une action de l'utilisateur. L'étiquette de la barre de titre de la boîte de message doit donc indiquer la source du message. N'utilisez pas de textes descriptifs du type « avertissement » ou « attention » : les boîtes de message comportent des symboles graphiques qui communiquent la nature du message, tel qu'indiqué à la figure suivante :

Figure 28 : Boîte de message



5.3.2 Types de boîtes de message

Généralement, les boîtes de messages comportent un symbole graphique qui indique le type du message affiché. La plupart des messages font partie de l'une des catégories mentionnées dans le tableau qui suit :

Tableau 3 : Types de messages et symboles associés

Symbole	Type de message	Description
	Avertissement	Avertit l'utilisateur d'une situation qui requiert une information ou une décision de la part de l'utilisateur avant d'exécuter une opération : par exemple, une action en attente ayant des conséquences potentiellement destructives et irréversibles. Le message peut être sous forme de question, par exemple : « Enregistrer les changements dans Montexte ? ».
	Problème critique	Informe l'utilisateur de l'existence d'un problème sérieux qui exige une intervention ou une correction avant que le traitement ne puisse continuer. Par exemple, si l'utilisateur demande accès au lecteur de disquette et que ce dernier est vide, le message indiquera : « Le lecteur sélectionné n'est pas en service. Vérifiez qu'un disque est bien inséré. »
	Information	Donne l'information à propos des résultats d'une commande. En général, n'offre aucun choix à l'utilisateur. Ce dernier accepte le message en appuyant sur le bouton de commande OK.

Parce que l'affichage des boîtes de message interrompt la tâche courante de l'utilisateur, il est préférable d'afficher la boîte de message seulement lorsque la fenêtre de l'application qui l'affiche est active. N'affichez qu'une seule boîte de message pour alerter l'utilisateur dans une situation donnée. L'affichage de boîtes de message en chaîne tend à semer la confusion chez les utilisateurs.

5.3.3 Textes de boîte de message

Le texte de la boîte de message devrait être clair, concis et formulé dans des termes que l'utilisateur peut comprendre aisément. Il faut éviter d'utiliser des termes de jargon technique ou de l'information propre au système.

Suivez également ces lignes directrices pour vos textes de message :

- Formulez le problème, sa cause probable (si c'est possible) et ce que l'utilisateur peut faire, même lorsque la solution semble très évidente.
- Envisagez, lorsque cela est possible, le fait d'offrir la solution en tant qu'option offerte dans le message.
- Évitez d'employer inutilement des phrases trop complexes ou de la terminologie technique.
- Évitez les tournures de phrase qui blâment l'utilisateur ou sous-entendent une erreur de sa part. Évitez d'utiliser le mot *erreur*.
- Rendez les messages aussi spécifiques que possible. Évitez d'associer plus de deux ou trois conditions dans un même message : offrez plutôt un message pour chaque condition.

- Évitez de vous fier aux messages par défaut du système, tels que les messages d'erreur de MS-DOS® ; fournissez plutôt vos propres messages partout où cela est possible.
- Soyez bref mais explicite. Ne donnez que l'information contextuelle nécessaire. On recommande généralement de limiter la longueur des messages à deux ou trois lignes de texte. Si plus d'informations sont nécessaires, il est possible d'y donner accès par un bouton de commande ouvrant une fenêtre d'aide.

6. GESTION DE L'ENVIRONNEMENT FENÊTRÉ

Les tâches des utilisateurs impliquent souvent de travailler avec des informations de différents types, contenues dans plus d'une fenêtre ou plus d'un affichage. Il y a plusieurs techniques pour gérer un ensemble de fenêtres ou d'affichages. Cette section aborde certaines des techniques les plus répandues ainsi que des facteurs à considérer pour choisir un modèle en particulier.

6.1 INTERFACE À DOCUMENT UNIQUE

Dans plusieurs cas, l'interface d'un objet ou d'une application peut être contenue dans une seule fenêtre principale, augmentée d'un ensemble de fenêtres secondaires. La gestion des fenêtres primaires est effectuée par l'entremise du *Bureau* et de la fenêtre *Liste de tâches*. À son ouverture, la fenêtre est placée au sommet de l'ordre en Z et son activation est indiquée dans la fenêtre *Liste de tâches* ; cela facilite le passage d'une fenêtre à l'autre, sans avoir à déplacer les fenêtres pour y accéder.

L'utilisation d'un modèle à instance unique (*Single document interface* – SDI), où une seule fenêtre principale est activée, rend la gestion des fenêtres plus simple et réduit les chances de confusion pour l'utilisateur.

De plus, OLE supporte la création de documents composés ou d'autres types de contenants à documents. À l'aide de ces constructions, l'utilisateur peut assembler un ensemble d'objets différents dans un but spécifique à l'intérieur d'une même fenêtre principale, éliminant ainsi la nécessité d'afficher ou d'éditer l'information dans des fenêtres séparées.

Certains types d'objets, tels que des objets associés à des dispositifs (souris, imprimante, etc.), n'ont parfois même pas besoin de fenêtre principale et n'utilisent qu'une fenêtre secondaire pour permettre la visualisation et l'édition de leurs propriétés.

Il est également possible pour un objet de n'avoir aucune fenêtre : une icône est sa seule représentation. Dans ce cas très rare, il faut fournir un ensemble de commandes menu appropriées afin de permettre à l'utilisateur de contrôler l'activité de cet objet.

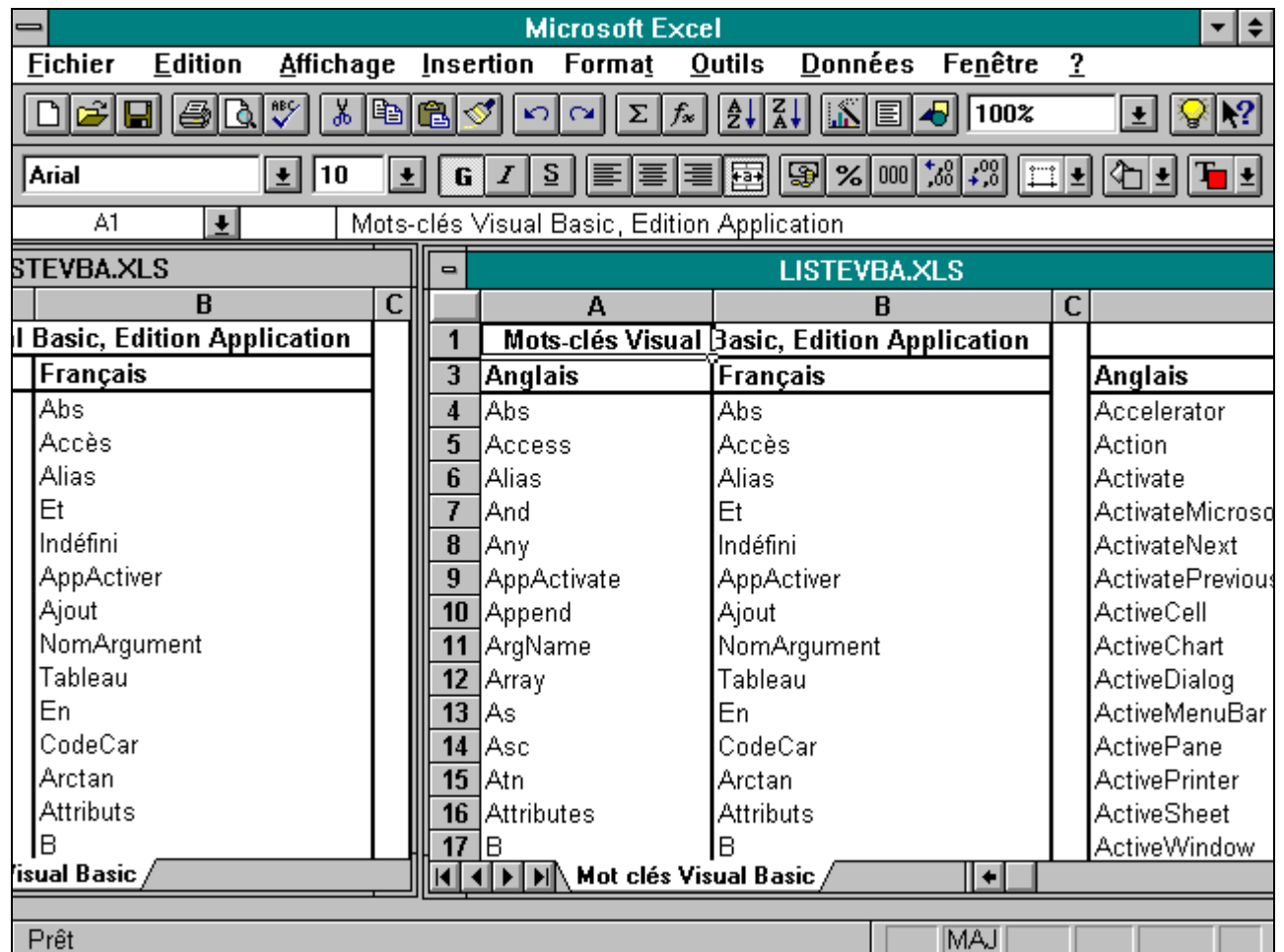
6.2 INTERFACE À DOCUMENTS MULTIPLES

Pour certaines tâches, le *Bureau* et la *Liste des tâches* ne suffisent plus à la gestion d'un ensemble de fenêtres : par exemple, il peut être plus efficace de présenter des vues multiples des mêmes données ou de données reliées dans des fenêtres partageant des éléments d'interface. Dans ce genre de situation, utilisez une interface à documents multiples (*Multiple document interface* – MDI).

La technique MDI utilise une fenêtre principale unique, appelée fenêtre parent, qui contient visuellement un ensemble de documents associés, ou fenêtres enfants, tel qu'illustré à la figure suivante. Chaque fenêtre enfant est essentiellement une fenêtre principale, mais n'apparaît qu'à l'intérieur de la fenêtre parent et non sur le *Bureau*. La fenêtre parent offre à ses fenêtres enfants une structure visuelle et opérationnelle. Par exemple, les fenêtres enfants partagent généralement la barre de menu de leur fenêtre parent et

peuvent également partager d'autres éléments de son interface, tels que des barres d'outils ou des barres d'état.

Figure 29 : Fenêtre parent et enfant MDI



Les fenêtres secondaires (boîtes de dialogue, boîtes de message, feuilles de propriétés, etc.) affichées à la suite d'une interaction dans une fenêtre MDI, parent ou enfant, ne sont généralement pas contenues ou coupées (*Clipped*) dans la fenêtre parent. Ces fenêtres devraient être activées et affichées selon les conventions qui s'appliquent pour les fenêtres secondaires associées à des fenêtres principales, même si elles s'appliquent à des fenêtres enfants individuelles. Une application MDI s'active de la même manière qu'une autre application. La fenêtre parent s'ouvre d'abord et les fenêtres enfants d'un fichier donné s'ouvrent ensuite à l'intérieur.

Si l'utilisateur ouvre directement un document MDI à l'extérieur de l'interface de sa fenêtre parent (ex. : en double-cliquant sur le fichier dans le *Gestionnaire de fichiers*) et que la fenêtre parent est déjà ouverte, il lui faut alors ouvrir une nouvelle instance de la fenêtre parent. Bien que l'ouverture de la fenêtre enfant à l'intérieur de la fenêtre parent existante puisse être plus efficace, l'ouverture d'un nouveau document peut affecter l'environnement de la tâche déjà établi dans cette fenêtre parent. Si l'utilisateur souhaite ouvrir un document à l'intérieur d'une fenêtre parent donnée, il peut le faire grâce aux commandes contenues dans cette fenêtre.

Comme les fenêtres enfants MDI sont des fenêtres principales, leur fermeture s'effectue en suivant les mêmes conventions que pour les fenêtres principales, soit en incluant un bouton *Fermer* dans leur barre de titre et un bouton de commande *Fermer* dans le menu autonome pour la fenêtre. Lorsque l'utilisateur ferme toutes les fenêtres enfants, ne fermez pas la fenêtre parent, à moins que cette dernière ne puisse fournir de contexte ou d'opérations sans fenêtre enfant ouverte. Lorsque l'utilisateur ferme la fenêtre parent, fermez toutes les fenêtres enfants associées.

La technique MDI permet à l'utilisateur de déplacer ou de cacher les fenêtres enfants en déplaçant ou en minimisant la fenêtre parent. Lorsque l'utilisateur déplace une fenêtre parent MDI, conservez les positions relatives des fenêtres enfants ouvertes à l'intérieur de la fenêtre parent. Le déplacement des fenêtres enfants est contraint par la fenêtre parent. Dans certains cas, la taille de la fenêtre parent oblige à couper une fenêtre enfant. Lorsque l'utilisateur maximise une fenêtre parent, ou une fenêtre enfant MDI, augmentez la fenêtre à sa taille maximale. Lorsque cette taille d'une fenêtre enfant outrepassa la taille de sa fenêtre parent, fusionnez la fenêtre enfant avec la fenêtre parent.

Les conventions d'activation et de navigation entre les fenêtres enfants sont les mêmes que pour les fenêtres principales. Le menu de la fenêtre parent devrait comporter des commandes pour basculer entre les fenêtres enfants ou pour les gérer et les arranger (ex. : en mosaïque ou en cascade). Lorsque l'utilisateur bascule entre les fenêtres enfants, il est possible de modifier l'interface de la fenêtre parent (barre de menu, barre d'outils ou barre d'état) afin de refléter les commandes propres à une fenêtre enfant donnée. Toutefois, il est important de maintenir la cohérence de l'interface.

7. MENUS, COMMANDES ET BARRES D'OUTILS

L'environnement Windows offre plusieurs composantes interactives qui facilitent la création d'interfaces pour exécuter des commandes et spécifier des valeurs. Ces composantes fournissent également une structure cohérente et un ensemble de conventions d'interface. Cette section décrit les éléments interactifs qui composent les menus, les commandes et les barres d'outils et précise la manière de les utiliser.

7.1 MENUS

Les *menus* présentent la liste des commandes disponibles à l'utilisateur.

Il existe plusieurs types de menus, tels que des menus déroulants, des menus autonomes ou des menus en cascade. La section suivante présente ces menus de manière plus détaillée.

7.1.1 Barre de menu et menus déroulants

La barre de menu est l'une des formes les plus répandues de menus. Il s'agit d'une zone affichée au sommet d'une fenêtre, située directement sous la barre de titre, telle qu'illustrée à la figure suivante. Une barre de menu comprend un ensemble d'inscriptions appelées *titres de menu*. Chaque titre de menu donne accès à un *menu déroulant* composé d'un groupe d'*éléments de menu* (ou choix).

Figure 30 : Barre de menu



Le contenu d'une barre de menu et des menus déroulants qui l'accompagnent est déterminé par les fonctionnalités de votre application et par le contexte de l'interaction avec l'utilisateur. Vous pouvez permettre à l'utilisateur de configurer la structure de la barre de menu et même l'autoriser à la cacher entièrement ; donnez-lui alors accès aux fonctionnalités normalement fournies par la barre de menu par l'entremise de menus autonomes, de poignées et de barres d'outils que vous ajoutez à l'interface.

Un menu déroulant, lorsqu'il est affiché, apparaît sous la forme d'un panneau où les éléments du menu sont organisés en colonne. Évitez d'afficher plus d'une colonne d'éléments de menu.

Interaction des menus déroulants

Lorsque l'utilisateur choisit le titre d'un menu, ce dernier affiche le menu déroulant associé. Pour afficher un menu déroulant à l'aide de la souris, l'utilisateur pointe sur le titre du menu et appuie sur le bouton 1. Cette action ouvre le menu et surligne son titre.

Si l'utilisateur ouvre un menu en appuyant sur le bouton de la souris alors que le pointeur est sur le titre du menu, il peut glisser le pointeur sur l'un des éléments du menu déroulant tout en maintenant le bouton de la souris enfoncé. Lorsque l'utilisateur glisse le pointeur, les éléments sont successivement surlignés, indiquant la position du pointeur pendant son déplacement. L'utilisateur sélectionne la commande associée à un élément de menu lorsqu'il relâche le bouton de la souris, au moment où le pointeur est situé au-dessus de cet élément. Le système retire alors le menu déroulant. Si l'utilisateur déplace le pointeur hors de la zone du menu avant d'avoir effectué une sélection et relâche le bouton de la souris, le menu est « annulé » et le menu déroulant est retiré de l'écran. Toutefois, si l'utilisateur ramène le pointeur sur la zone de menu avant d'avoir relâché le bouton, il peut encore choisir un élément.

Si l'utilisateur ouvre un menu en cliquant sur le titre du menu, ce dernier est alors surligné et le menu déroulant demeure affiché jusqu'à ce que l'utilisateur clique à nouveau sur la souris. L'utilisateur peut choisir un élément de commande d'un menu déroulant de deux manières : en cliquant sur un des éléments du menu ou en glissant jusque sur l'élément et en relâchant le bouton. Le menu déroulant est alors retiré de l'écran. Cliquer à nouveau sur le titre d'un menu affiché fait disparaître ce menu. De même, cliquer sur le titre d'un autre menu fait disparaître le menu déroulant du premier pour faire afficher le menu déroulant du second.

L'interface clavier pour les menus déroulants utilise la touche ALT pour activer les éléments de la barre de menu. Lorsque l'utilisateur appuie sur une touche alphanumérique en maintenant la touche ALT enfoncée (ou après qu'il a relâché la touche ALT), le système affiche le menu déroulant dont la touche d'accès rapide correspond à la touche alphanumérique enfoncée. Lorsque le menu déroulant est affiché, l'utilisateur peut choisir l'un des éléments en appuyant sur la touche alphanumérique qui correspond à sa touche d'accès rapide.

L'utilisateur peut également utiliser les touches fléchées pour accéder aux menus déroulants par le clavier. Lorsque l'utilisateur appuie sur ALT mais n'a pas encore sélectionné un menu déroulant, les touches FLÈCHE GAUCHE et FLÈCHE DROITE surlignent respectivement le menu précédent ou le menu suivant. À l'extrémité de la barre de menu, les flèches permettent de se déplacer en boucle pour surligner le premier élément à l'autre extrémité de la barre de menu. Pour afficher le menu déroulant associé à un élément surligné, l'utilisateur doit appuyer sur la touche ENTRÉE. Si un menu déroulant est déjà affiché sur une barre de menu, les touches FLÈCHE GAUCHE et FLÈCHE DROITE permettent de naviguer jusqu'au prochain menu déroulant dans la direction donnée. Si un menu déroulant affiche son contenu en plusieurs colonnes, les touches fléchées permettent alors de naviguer à l'intérieur de ce menu ; à la fin de ce dernier, les flèches permettent alors de passer au prochain menu.

L'utilisation des touches FLÈCHE HAUT et FLÈCHE BAS permet également d'afficher un menu déroulant si aucun n'est ouvert. Dans un menu déroulant ouvert, ces touches permettent de se déplacer parmi les éléments de ce menu, surlignant les éléments parcourus et bouclant aux extrémités. Si le menu est réparti sur plusieurs colonnes, les touches fléchées bouclent (*wrap*) d'une colonne à l'autre.

L'utilisateur peut annuler l'affichage d'un menu déroulant en appuyant sur la touche ALT lorsque la barre de menu est active. Cette action permet non seulement de retirer le menu déroulant mais également de désactiver la barre de menu. Appuyer sur la touche ESC annule également l'affichage d'un menu déroulant, mais seulement à ce niveau. Cela signifie que si un menu déroulant est ouvert et que l'utilisateur appuie sur la touche ESC, ce menu disparaît de l'écran mais son titre demeure surligné. En appuyant sur

ESC une seconde fois, le titre du menu redevient normal et la barre de menu est désactivée : le contenu de la fenêtre devient alors le centre d'attention.

En ce qui a trait à l'attribution des raccourcis au clavier aux commandes des menus déroulants, la commande est immédiatement exécutée lorsque l'utilisateur appuie sur ces touches. De manière facultative, il est également possible de surligner le titre du menu contenant la commande activée avec les raccourcis au clavier ; toutefois, il ne faut pas afficher le menu déroulant associé.

7.1.2 Menus déroulants communs

Cette section décrit les conventions d'utilisation pour les menus déroulants les plus couramment utilisés dans les applications. Ces menus ne sont pas requis pour tous les types d'application mais il est recommandé d'employer ces lignes directrices pour implanter ces types de menu dans votre interface.

Menu *Fichier*

Figure 31 : Exemple d'un menu *Fichier*

<u>N</u> ouveau...	Ctrl+N
<u>O</u> uvrir...	Ctrl+O
<u>E</u> nregistrer	Ctrl+S
Enregistrer sous...	
<u>I</u> mprimer...	Ctrl+P
Aperçu avant impression	
<u>M</u> ise en page...	
<u>1</u> C:\WINDOWS\...\xolo.xls	
<u>2</u> D:\m262xsfa.doc	
<u>3</u> D:\m262xsfa.rtf	
<u>4</u> D:\M262xsfa.wri	
Envoyer...	
<u>Q</u> uitter	

Le menu *Fichier* est l'interface pour les opérations de base qui s'appliquent aux fichiers. Votre application devrait permettre les opérations *Ouvrir*, *Enregistrer*, *Envoyer vers* et *Imprimer*.

Si votre application autorise la commande *Quitter*, placez-la au bas du menu *Fichier*, précédée d'un séparateur de menu. Lorsque l'utilisateur choisit la commande *Quitter*, fermez toutes les fenêtres ouvertes et tous les fichiers actifs et interrompez tous les traitements en cours. S'il s'agit d'un objet qui demeure actif même lorsque sa fenêtre est fermée, tel un fichier ou une imprimante, utilisez alors la commande *Fermer* plutôt que la commande *Quitter*.

Menu *Edition*

Figure 32 : Exemple d'un menu *Edition*

<u>A</u> nnuler Mettre à jour les champs	Ctrl+Z
Répéter Mettre à jour les champs	Ctrl+Y
<u>C</u> ouper	Ctrl+X
Copier	Ctrl+C
Coller	Ctrl+V
Collage spécial...	
Effacer	Suppr
Sélectionner tout	Ctrl+A
Rechercher...	Ctrl+F
Remplacer...	Ctrl+H
Atteindre...	Ctrl+B
Insertion automatique...	
Signet...	
<u>L</u> iaison...	
<u>I</u> mage	

Dans le menu *Edition*, placez les commandes générales d'édition. Ces commandes comprennent les commandes de transfert *Couper*, *Copier* et *Coller*, les commandes pour les objets OLE, ainsi que les commandes suivantes (si votre application les autorise) :

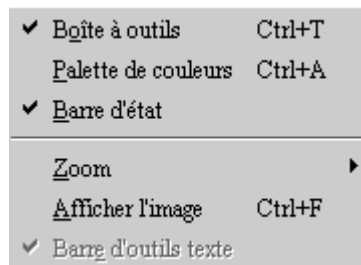
Commande	Fonction
Annuler	Défait la dernière action exécutée.
Répéter	Répète la dernière action exécutée.
Rechercher et remplacer	Cherche du texte et le remplace dans le document.
Effacer	Retire la sélection active.
Dupliquer	Crée une copie de la sélection active.

Placez ces commandes dans le menu *Edition* ainsi que dans le menu autonome de l'objet sélectionné.

Note : Ne mettez pas l'accent à *Edition* afin que la touche d'accès rapide correspondant à ce menu soit cohérente avec la plupart des applications développées sous Windows.

Menu *Affichage*

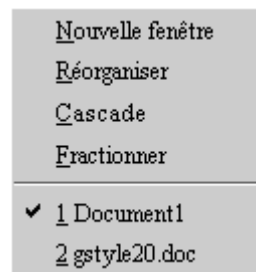
Figure 33 : Exemple d'un menu *Affichage*



Dans ce menu, placez les commandes qui modifient l'affichage des données dans la fenêtre active. Inclure les commandes qui modifient l'affichage des données et non pas les données elles-mêmes, tel que Zoom ou Plan. Inclure également les commandes qui permettent de contrôler l'affichage de certains éléments de l'interface, tel que *Afficher règle*. Ces commandes devraient également se trouver dans le menu autonome de la fenêtre ou du panneau.

Menu *Fenêtre*

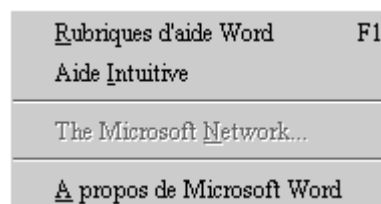
Figure 34 : Exemple d'un menu *Fenêtre*



Utilisez le menu *Fenêtre* dans le cadre d'applications à interface à multiples documents (ou MDI – *Multiple Document Interface*) pour les commandes associées à la gestion des fenêtres de l'espace de travail : *Nouvelle fenêtre*, *Mosaïque*, *Cascade*, etc. Inclure également ces commandes dans le menu autonome de la fenêtre parent de l'application MDI.

Menu ? (*Aide*)

Figure 35 : Exemple d'un menu ?



Utilisez le menu ? pour les commandes qui permettent l'accès à l'information de l'*Aide*. Inclure les commandes suivantes : *Index*, *Rechercher l'Aide sur*.

Si vous donnez accès aux informations de droits d'auteur et de version de votre application, inclure une commande *À propos de Votre Application* dans ce menu. Lorsque l'utilisateur choisit cette commande, affichez une fenêtre qui présente le nom de l'application, le numéro de version, les informations de droits d'auteur et toute autre propriété associée. Affichez cette information dans une boîte de dialogue ou encore dans une page de droits d'auteur sur la feuille de propriété du fichier principal de l'application. N'ajoutez pas d'ellipse (...) à la fin de cette commande puisque le système n'exige pas de paramètres additionnels de la part de l'utilisateur.

7.1.3 Menus en cascade

Un *menu en cascade* (également appelé *menu hiérarchique* ou menu enfant) est un sous-menu d'un élément de menu. L'indice visuel de la présence d'un menu en cascade est l'inclusion d'une flèche triangulaire affichée à côté de l'étiquette de l'élément de menu parent.

Utilisez les menus en cascade pour fournir à l'utilisateur un accès à des choix additionnels, plutôt que de prendre davantage d'espace dans le menu parent, ou pour afficher des objets liés par une relation hiérarchique.

Prenez garde aux désavantages des menus en cascade : ils augmentent la complexité de l'interface menu en forçant l'utilisateur à naviguer plus en profondeur dans la structure du menu pour atteindre un choix particulier. Ils exigent également plus de coordination de la part de l'utilisateur pour maîtriser les changements de direction nécessaires pour naviguer à l'intérieur. Utilisez donc les menus en cascade avec modération. Minimisez le nombre de niveaux pour les éléments de menu, en limitant idéalement votre conception à un seul sous-menu. Évitez d'utiliser les menus en cascade pour des commandes répétitives ou fréquentes.

En guise d'alternative, rendez les choix accessibles à partir d'une fenêtre secondaire, spécialement lorsqu'il s'agit de réglages indépendants : cela permet à l'utilisateur de régler de nombreuses options en invoquant une seule commande. Il est également possible de supporter plusieurs options répandues en tant que commandes sur une barre d'outils.

L'interaction de l'utilisateur avec un menu en cascade est similaire à celle d'un menu déroulant de la barre de menu, sauf que le menu en cascade ne s'affiche qu'après un bref délai. Cela évite l'affichage inutile du menu si l'utilisateur est en train de fureter ou de naviguer d'un élément de menu à l'autre. Une fois affiché, si l'utilisateur déplace le pointeur sur un autre élément du menu, le menu en cascade est retiré de l'écran après un bref délai. Ce délai permet à l'utilisateur de glisser directement du menu parent à un élément de son menu en cascade.

7.1.4 Titres de menu

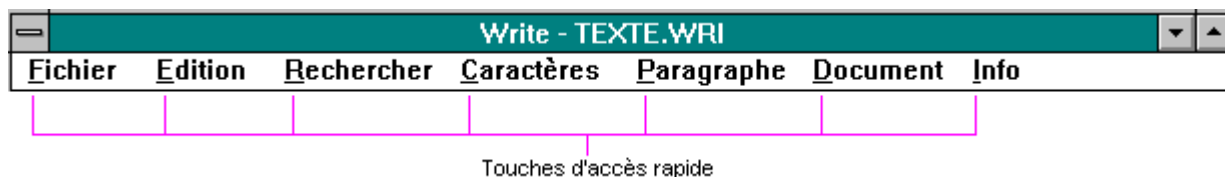
Tous les menus déroulants et les menus en cascade possèdent un titre de menu. Pour les menus déroulants, le titre de menu apparaît sur la barre de menu ; pour les menus en cascade, le titre du menu est le nom de l'élément du menu parent. Les titres de menu sont représentatifs du menu entier et devraient communiquer aussi clairement que possible le but de tous les éléments du menu.

Autant que possible, n'utilisez qu'un seul mot pour titrer un menu de barre de menu. Des titres comportant plus d'un mot ou comportant des espaces sont difficiles à distinguer de titres à mot unique. De plus, évitez les mots composés peu répandus tels que *Taille-Caractère*.

Définissez l'un des caractères de chaque titre de menu en guise de touche d'accès rapide. Ce caractère fournit un accès clavier au menu. Windows affiche le raccourci au clavier d'un titre de menu en soulignant le caractère en question, tel qu'illustré à la figure suivante :

Note : Pour plus d'information sur l'entrée de données par le clavier et la définition des touches d'accès rapide, voir la section 2 – dispositifs d'interaction.

Figure 36 : Touches d'accès rapide dans une barre de menu



Ne définissez pas une même touche d'accès rapide pour plus d'un titre de menu.

7.1.5 Éléments de menu

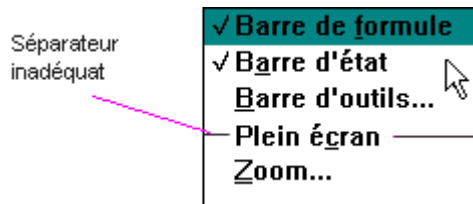
Les éléments de menu sont les choix individuels qui apparaissent dans un menu. Les éléments de menu peuvent être textuels, graphiques (tels que des icônes) ou une combinaison des deux et représentent les actions présentées dans le menu. Le format d'un élément de menu fournit à l'utilisateur un indice visuel sur la nature de l'action qu'il représente, tel qu'illustré à la figure suivante :

Figure 37 : Formats pour différents éléments de menu

Impossible d'annuler	Ctrl+Z
Répéter Effacer	F4
Couper	Ctrl+X
Copier	Ctrl+C
Coller	Ctrl+V
Collage spécial...	
Recopier	▶
Effacer	▶
Supprimer...	
Supprimer une feuille	
Déplacer ou copier une feuille...	
Rechercher...	Ctrl+F
Remplacer...	Ctrl+H
Atteindre...	F5
Liaison...	
Objet	

Lorsqu'un menu contient un ensemble d'éléments associés, séparez cet ensemble à l'aide d'une ligne de regroupement appelée *séparateur*. Le séparateur standard est une ligne simple qui s'étend sur toute la largeur du menu. Il faut éviter d'utiliser les éléments de menu eux-mêmes en guise de séparateurs, tel qu'illustré à la figure suivante :

Figure 38 : Séparateur inadéquat



Offrez toujours à l'utilisateur un indice visuel lui indiquant quels sont les éléments de menu qui peuvent être utilisés. Si un élément de menu est inadéquat ou inapplicable dans un contexte donné, rendez-le inactif ou retirez-le du menu. Le fait de laisser un élément de menu non valide actif puis de présenter une boîte de message si l'utilisateur le choisit est un retour d'information médiocre.

En général, il est préférable de désactiver un élément de menu plutôt que de le retirer car cela confère plus de stabilité à l'IPS. Toutefois, si le contexte est tel que l'élément de menu n'est plus pertinent, retirez-le. Par exemple, si un menu affiche un ensemble de fichiers ouverts et que l'un de ces fichiers est fermé ou effacé, il est approprié de retirer l'élément de menu correspondant.

Si tous les éléments d'un menu sont désactivés, désactivez le titre du menu. La désactivation d'un élément de menu (ou son titre) n'empêche pas l'utilisateur de fureter ou de choisir ce menu. Si vous présentez des messages sur la barre d'état, affichez un message indiquant que la commande n'est pas disponible et quelle en est la raison.

Le système fournit une apparence standard pour afficher les éléments de menu désactivés. Si vous fournissez vos propres images pour un élément de menu désactivé, suivre les lignes directrices usuelles sur la manière d'afficher un objet désactivé.

Types d'éléments de menu

Plusieurs éléments de menu prennent effet dès le moment où ils sont choisis. Si l'élément de menu est une commande qui requiert de l'information additionnelle pour compléter son exécution, ajouter une ellipse (...) à la fin de la commande. L'ellipse informe l'utilisateur qu'il manque de l'information et qu'il aura à fournir de l'information additionnelle. De telles commandes entraînent généralement l'affichage d'une boîte de dialogue. Par exemple, la commande *Enregistrer sous* comprend une ellipse car la commande n'est pas complète tant que l'utilisateur n'a pas fourni ou confirmé un nom de fichier.

Ce ne sont pas toutes les commandes affichant une boîte de dialogue (ou une autre fenêtre secondaire) qui doivent inclure une ellipse. Par exemple, il ne faut pas ajouter d'ellipse à la commande *Propriétés* simplement parce que le fait d'activer cette commande amène l'affichage d'une fenêtre secondaire : après l'activation de la commande, il n'y a plus de paramètres ou d'actions nécessaires pour atteindre le but de cette fenêtre. De même, on ne doit pas inclure d'ellipse pour une commande qui pourrait éventuellement faire afficher une boîte de message.

Alors qu'il est possible d'utiliser des éléments de menu pour exécuter des commandes, vous pouvez également les utiliser pour passer d'un mode à un autre, régler un état ou une propriété, plutôt que d'amorcer un traitement. Par exemple, le choix d'un élément dans un menu qui contient une liste d'outils ou d'affichage implique le changement vers cet état. Si l'élément de menu représente la valeur d'une propriété, lorsque l'utilisateur choisit cette propriété, sa valeur change.

Les éléments de menu permettant le réglage des états des objets peuvent être indépendants ou interdépendants.

- Les réglages indépendants sont les équivalents menu des cases à cocher. Par exemple, si un menu contient des propriétés de texte, telles que *Gras* ou *Italique*, ils forment un groupe de réglages indépendants. L'utilisateur peut changer chaque réglage sans affecter les autres, bien qu'ils s'appliquent tous deux à une sélection de texte unique. Inclure un crochet (✓) à gauche d'un réglage indépendant lorsque cet état s'applique.
- Les réglages interdépendants sont l'équivalent menu des boutons à option. Par exemple, si un menu contient des propriétés de justification (telles que *Gauche*, *Droite* ou *Centrée*), elles forment un groupe de réglages interdépendants. Parce qu'un paragraphe donné ne peut avoir qu'un seul type de justification, le choix d'un type de justification règle la propriété à cette valeur. Lorsque l'utilisateur choisit un élément interdépendant, placer un marqueur de bouton à option (●) à la gauche de l'élément sélectionné.

Lorsque vous utilisez le menu pour représenter les deux états d'un réglage, si ces deux états sont de toute évidence des contraires (tels que la présence ou l'absence d'une propriété), il est possible d'utiliser un crochet pour indiquer à quel moment le réglage s'applique. Par exemple, pour indiquer l'état d'une sélection de texte avec un élément de menu étiqueté *Gras*, affichez un crochet du côté gauche de l'élément lorsque la sélection est grasse et retirez-la lorsqu'elle ne l'est pas.

Étiquettes des éléments de menu

Utilisez un texte descriptif ou une étiquette graphique pour chaque élément de menu. Même si vous fournissez un graphique pour l'étiquette, envisagez également l'emploi de texte. Le texte permet un accès clavier plus direct ; de même, un plus grand nombre d'utilisateurs seront en mesure de comprendre le but de l'élément de menu.

Suivez les recommandations suivantes pour la conception d'étiquettes de menu :

- Donnez un nom unique aux éléments d'un même menu. Toutefois, les noms peuvent se répéter dans des menus différents pour représenter des actions similaires, ou même différentes.
- Utilisez un mot ou plus, mais surtout soyez bref.
- Définissez une touche d'accès rapide pour chaque élément d'un même menu. Cela offre un accès direct au menu pour l'utilisateur. Les recommandations pour le choix de la touche d'accès rapide sont les mêmes que pour les titres de menu, sauf que la touche d'accès rapide peut également être un chiffre placé au début du nom de l'élément. Cette technique est très utile, surtout pour les éléments variables, tels que des noms de fichiers.
- Suivez la convention usuelle pour les titres de livre, à savoir une majuscule à la première lettre du premier mot de la phrase ainsi que pour les noms propres.
- Évitez de changer les caractéristiques des polices de caractères pour un élément de menu individuel. Bien que ces caractéristiques illustrent un style particulier, elles peuvent également rendre le menu embrouillé ou illisible.

Raccourci au clavier des éléments de menu

Si vous associez un raccourci au clavier avec une commande, affichez le raccourci dans le menu ; alignez tous les raccourcis d'un même menu. Justifier à gauche à partir du premier taquet de tabulation situé après l'élément le plus long de la liste possédant un raccourci. N'utilisez pas d'espaces pour effectuer la justification car ils pourraient être mal affichés en fonction de la police de caractères utilisée par le système.

Afficher les raccourcis au clavier sous la forme suivante :

Avec la touche Contrôle :	CTRL+ <i>Touche</i>
Avec la touche Majuscule :	MAJ+ <i>Touche</i>
Avec la touche Alternner :	ALT+ <i>Touche</i>

Note : Pour plus d'information à propos des raccourcis au clavier, voir la section 2 – dispositifs d'interaction.

Évitez d'employer des raccourcis au clavier dans les menus autonomes.

7.2 CONTRÔLES

Les contrôles sont des objets graphiques qui représentent les propriétés ou les opérations associées à d'autres objets. Certains contrôles affichent des valeurs particulières et permettent leur édition. D'autres contrôles amorcent la commande associée.

Chaque contrôle possède une apparence et un comportement qui lui sont propres et qui permettent une forme spécifique d'interaction. Le système supporte également la conception de vos propres contrôles. Si vous définissez vous-même vos contrôles, adoptez des conventions cohérentes avec celles offertes par les contrôles du système.

Comme la plupart des éléments de l'interface, les contrôles fournissent un retour d'information indiquant à quel moment ils ont le centre d'attention ou sont activés. Par exemple, lorsque l'utilisateur interagit avec un contrôle à l'aide d'une souris, le contrôle indique qu'il est sélectionné à partir du moment où l'utilisateur enfonce le bouton de la souris ; le contrôle n'est activé que lorsque l'utilisateur relâche le bouton de la souris, sauf si le contrôle supporte l'autorépétition.

La plupart des contrôles portent des étiquettes qui identifient leur but : l'action dont ils assurent l'exécution ou encore la propriété qu'ils permettent de fixer. Étiquetez toujours les contrôles avec lesquels l'utilisateur doit interagir. Si un contrôle ne comporte pas d'étiquette, vous pouvez en attribuer une à l'aide d'un champ de texte statique. Concevez les étiquettes des boutons de commande selon les mêmes conventions de majuscules que les étiquettes des menus. Si le contrôle est désactivé, affichez l'étiquette du contrôle sous son apparence inactive.

Les contrôles décrits dans cette section sont standard : les outils de développement les plus répandus gèrent efficacement tous les aspects associés à ces contrôles, tant du point de vue de l'apparence que du comportement.

7.2.1 Boutons

Les boutons amorcent des actions ou modifient des propriétés. Il y a trois types de boutons de base : les boutons de commande, les cases à option et les cases à cocher.

Boutons de commande

Un bouton de commande est généralement rectangulaire et comporte une étiquette (texte, graphique ou encore les deux), tel qu'illustré à la figure suivante :

Figure 39 : Boutons de commande



Les effets de l'activation d'un bouton sont immédiats en fonction du contexte. Par exemple, dans une barre d'outils, l'activation d'un bouton permet l'exécution de l'action associée. Dans une fenêtre secondaire, telle qu'une boîte de dialogue, l'activation d'un bouton peut amener une transformation dans la fenêtre ou encore appliquer une transformation puis fermer la fenêtre.

Bien que les boutons de commande servent généralement à amorcer une action, ils peuvent également servir à refléter un mode ou un état, un peu comme les cases à option. Pour montrer qu'un objet est dans un état donné, affichez le bouton sous son apparence « enfoncée ».

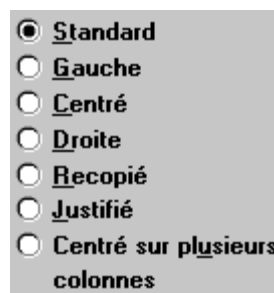
Figure 40 : Exemples d'apparence enfoncée et normale d'un bouton de commande



Cases à option

Une case à option, aussi appelée bouton radio, représente un choix unique parmi un ensemble de choix mutuellement exclusifs. En pratique, cela signifie que dans un groupe de cases à cocher, une seule case peut être choisie. Pour cette raison, rassemblez les cases à option en groupe de deux ou plus, tel qu'illustré à la figure suivante :

Figure 41 : Ensemble de cases à option



Les cases à option ont l'apparence d'un ensemble de petits cercles. Lorsque l'une des cases est choisie, un point noir apparaît au centre du cercle. Lorsque la case n'est pas choisie, le cercle est vide. Il n'est pas obligatoire de donner une option cochée par défaut. Il faut éviter d'utiliser les cases à option pour amorcer une action autre que celle de fixer une option ou une valeur représentée par la case à option. La seule exception consiste à autoriser le double-cliquage en guise de raccourci pour fixer une option puis exécuter la commande par défaut de la fenêtre où se trouve le contrôle, quand le choix d'une option est la principale utilité de la fenêtre.

Vous pouvez utiliser les cases à option pour représenter un ensemble de choix pour une propriété donnée. Limitez l'utilisation des cases à option pour de petits ensembles, généralement à sept ou moins, mais toujours au moins à deux. Si vous avez besoin de plus de choix, envisagez l'utilisation d'un autre contrôle, tel qu'une zone de liste à sélection unique.

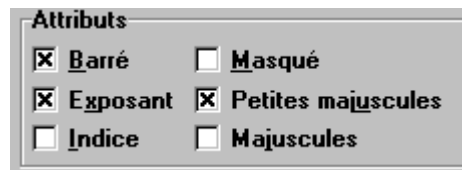
Parce que les cases à option apparaissent en groupe, il est possible de définir une boîte de contrôle pour définir visuellement le groupe de cases à option. Par exemple, pour la boîte de contrôle *Justification*, on retrouve les cases à option *Gauche*, *Centré*, *Droite*.

Cases à cocher

Comme les cases à option, les cases à cocher sont des options actives ou inactives ; les cases à cocher diffèrent des cases à options en ce qu'on les utilise pour faire des choix indépendants et non exclusifs. Comme pour les réglages indépendants dans les menus, n'utilisez les cases à cocher que lorsque les deux états du choix sont clairement opposés et non ambigus. Si ce n'est pas le cas, employez plutôt des cases à option ou une autre forme de contrôle à sélection unique.

Une boîte de sélection a l'apparence d'une boîte carrée assortie d'une étiquette. Lorsque la case est sélectionnée (cochée), un X apparaît dans la boîte. Lorsque la case n'est pas sélectionnée, la case à cocher est vide, telle qu'illustrée à la figure suivante :

Figure 42 : Ensemble de cases à cocher



Regroupez les choix de cases à cocher ensemble. Le regroupement des cases à cocher n'empêche pas l'utilisateur de cocher les cases selon la combinaison de son choix. Bien que chaque case à cocher soit généralement indépendante des autres, il est possible d'utiliser la sélection d'une case pour affecter les autres cases. Par exemple, si on choisit l'option *Exposant* (voir figure ci-dessus), alors l'option *Indice* sera forcément désactivée, puisque ces deux choix sont mutuellement exclusifs. On peut également faire l'inverse, à savoir que lorsque la sélection d'une option entraîne forcément une seconde, cette dernière sera sélectionnée automatiquement. Si vous avez un grand nombre de choix, ou si le nombre de choix varie, utilisez plutôt une liste à sélection multiple.

7.2.2 Zones de liste

Une zone de liste est un contrôle utilisé pour offrir une liste de choix à l'utilisateur. Les choix peuvent être du texte, des icônes, des couleurs ou d'autres graphiques. La zone de liste permet le choix d'un élément ou plus de la liste.

Les zones de liste sont plus appropriées pour afficher un grand nombre de choix qui varient en quantité ou en contenu. Par exemple, si une taille de caractère n'est pas disponible pour la police sélectionnée, ne pas l'afficher dans la liste.

Présentez les éléments de la liste selon l'ordre le plus approprié, de manière à faciliter la recherche des éléments par les utilisateurs. Par exemple, recourez à l'ordre alphabétique pour une liste de noms et à

l'ordre chronologique pour une liste de dates. S'il n'existe pas d'ordre logique donné pour une liste, utilisez l'ordre numérique croissant ou l'ordre alphabétique.

La largeur d'une zone de liste devrait être suffisante pour permettre l'affichage d'un élément de la liste de largeur moyenne. Si cette règle ne peut être mise en pratique à cause de l'espace disponible ou de la variabilité du contenu de la liste, envisagez l'une des options suivantes :

- Rendez la zone de liste suffisamment large pour que tous les éléments de la liste soient reconnaissables.
- Utilisez une ellipse (...) au centre ou à la fin d'un long élément pour le rendre plus court, tout en conservant les caractéristiques qui permettront de le reconnaître.
- Incorporez une barre de défilement horizontale. Toutefois, cette option réduit l'utilisabilité puisque l'ajout de la barre de défilement réduit le nombre d'éléments de la liste que l'utilisateur peut visualiser en un même coup d'œil.

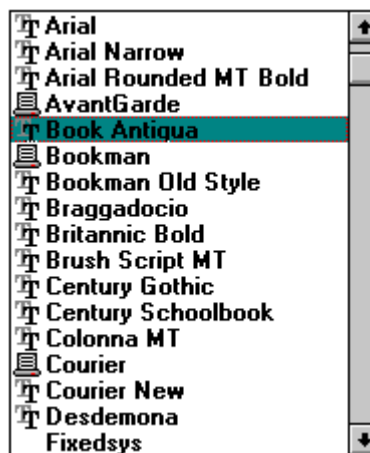
Lorsque l'utilisateur clique sur un élément de la zone de liste, cet élément est sélectionné. Le support pour les sélections multiples dépend du type de zone de liste que vous sélectionnez. Les zones de liste comportent également des barres de défilement vertical lorsque le nombre d'éléments de la liste dépasse les dimensions de la zone.

Les zones de liste peuvent être classées en fonction de leur type d'affichage et du type de sélection qu'elles permettent.

Liste à sélection unique

Les zones de liste à sélection unique sont conçues pour permettre la sélection d'un seul élément à la fois. Par conséquent, ce contrôle offre une opération de sélection mutuellement exclusive similaire à celle des cases à option, sauf qu'il permet de gérer plus efficacement un grand nombre d'éléments.

Figure 43 : Liste à sélection unique

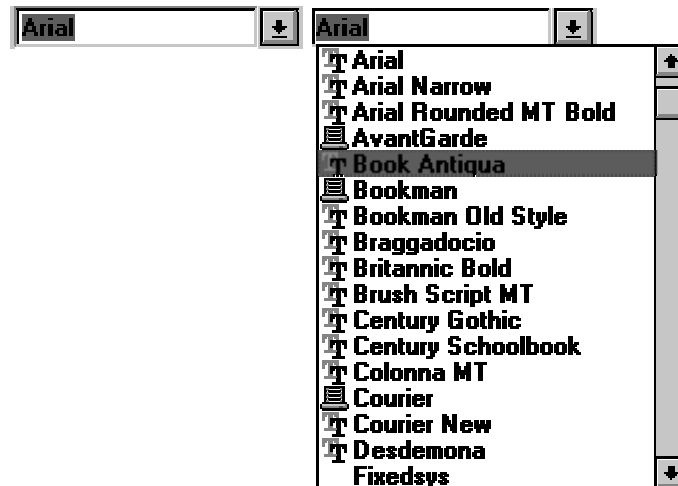


Liste déroulante

Tout comme la liste à sélection unique, une liste déroulante permet la sélection d'un seul élément parmi une liste ; elle diffère par le fait que la liste n'est affichée que sur demande. Dans son état « fermé », la

liste affiche la valeur courante de ce contrôle. L'utilisateur peut ouvrir la liste pour changer cette valeur. La figure qui suit montre une liste déroulante dans son état fermé et son état ouvert.

Figure 44 : Liste déroulante (états fermé et ouvert)



Bien que les listes déroulantes soient une méthode efficace pour sauver de l'espace et ainsi réduire l'encombrement, elles exigent davantage d'interaction de la part de l'utilisateur pour fureter et pour sélectionner un élément.

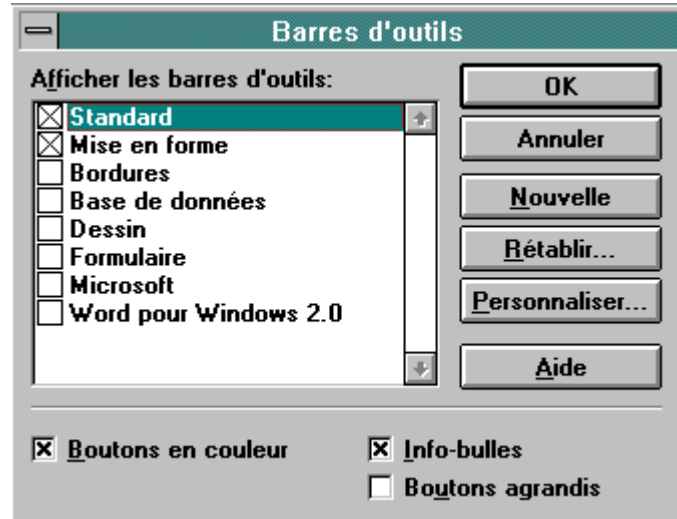
Liste à sélection étendue et à sélection multiple

Bien que la plupart des listes soient des listes à sélection unique, certains contextes exigent le choix de plus d'un élément. Les zones de liste à sélection étendue et les listes à sélection multiple permettent ces fonctionnalités. La conception de ces listes suit les mêmes conventions que pour les zones de liste à sélection unique.

Les listes à sélection étendue sont conçues pour permettre la sélection unique ou la sélection d'un ensemble d'éléments joints dans la liste. Elles ont la même apparence que les listes à sélection unique. On peut utiliser le clavier ou la souris pour sélectionner un ensemble. Avec le clavier, il faut appuyer sur la touche MAJ et la maintenir enfoncée puis utiliser les touches fléchées pour sélectionner l'ensemble ; avec la souris, appuyez sur le bouton gauche de la souris lorsque le pointeur est situé au-dessus du premier élément, puis sélectionnez l'ensemble en glissant. Les listes étendues permettent également la sélection d'éléments disjoints : en maintenant la touche CTRL enfoncée, cliquez sur les éléments de la liste désirés.

Quant aux listes à sélection multiple, elles sont optimisées pour permettre la sélection d'éléments disjoints dans la liste ; la sélection d'éléments joints est également permise. Afin de distinguer visuellement les deux types de liste, il est suggéré de concevoir la liste à sélection multiple sous la forme d'une liste défilante de cases à cocher, telle qu'illustrée à la figure qui suit :

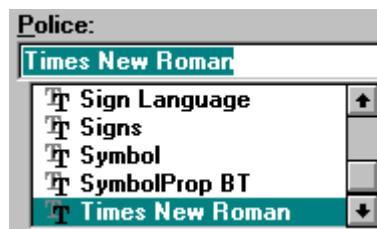
Figure 45 : Liste à sélection multiple



Liste modifiable

Une zone de liste modifiable est un contrôle combinant une zone de texte et une zone de liste, telle qu'illustrée à la figure suivante. Ce contrôle permet à l'utilisateur d'entrer une valeur au clavier ou de sélectionner l'un des éléments de la liste.

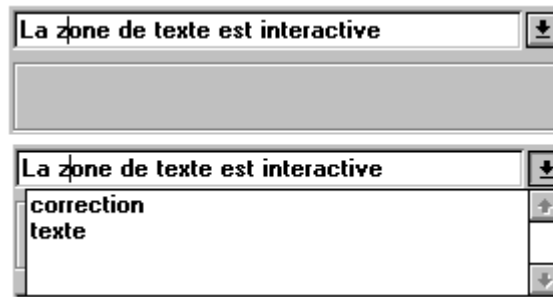
Figure 46 : Liste modifiable



La zone de texte et la liste sont reliées : au fur et à mesure que le texte est entré au clavier et s'inscrit dans la zone de texte, la liste défile vers l'élément qui s'approche le plus du texte entré. De plus, lorsque l'utilisateur sélectionne un élément dans la liste, la sélection remplace immédiatement le contenu de la zone de texte.

Liste déroulante modifiable

Ce contrôle combine les caractéristiques d'une zone de texte et d'une liste déroulante (voir figure suivante). Il est plus compact que la zone de liste modifiable et peut être utilisé pour sauver de l'espace ; par contre, il exige davantage d'interaction de la part de l'utilisateur pour afficher la liste.

Figure 47 : Liste déroulante modifiable (états fermé et ouvert)

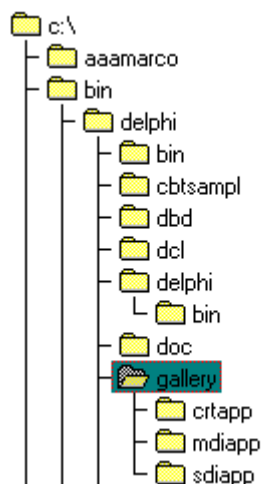
L'état fermé du contrôle est semblable à celui d'une liste déroulante, sauf que la zone de texte est interactive. La liste se déroule lorsque l'utilisateur appuie sur le bouton. Le texte entré par l'utilisateur est ajouté à la liste existante. La liste se referme dans trois cas : si l'utilisateur appuie à nouveau sur le bouton, s'il appuie sur un autre contrôle ou s'il sélectionne un élément de la liste.

Contrôles à liste d'icônes

Un contrôle à liste d'icônes est une liste à sélection étendue spéciale qui affiche un ensemble d'éléments, chaque élément étant constitué d'une icône et d'une étiquette. Utilisez ce contrôle là où il convient de représenter les objets sous forme d'icônes : cette technique offre un paradigme cohérent pour l'utilisateur habitué à interagir avec des icônes ailleurs dans l'IPS.

Contrôles à arborescence

Une arborescence est une liste spéciale affichant un ensemble d'objets en mode plan indenté en fonction d'une relation hiérarchique logique. Ce contrôle comporte des boutons qui permettent à l'arborescence de se déployer ou de se replier, tel qu'illustré à la figure suivante. Il est possible d'utiliser ce contrôle pour afficher les relations hiérarchiques entre des contenants (tels que des chemises) ou d'autres types d'éléments.

Figure 48 : Arborescence

7.2.3 Champs de texte

L'environnement Windows comprend plusieurs contrôles qui facilitent l'affichage, l'entrée et l'édition d'une valeur texte. Certains de ces contrôles combinent un champ d'entrée de texte avec d'autres types de contrôles.

Les étiquettes ne font pas partie du champ de texte. Toutefois, il est possible d'ajouter une étiquette en utilisant un champ de texte statique. L'ajout d'une étiquette aide à identifier le but du champ de texte et permet d'indiquer si le champ est activé ou non.

Zone de texte

Une zone de texte est un contrôle rectangulaire où l'utilisateur peut entrer ou éditer du texte, tel qu'illustré à la figure suivante. Cette zone peut être définie pour une ou plusieurs lignes de texte. Les bordures du contrôle sont optionnelles, bien qu'elles soient généralement incluses lorsque la zone de texte est affichée dans une barre d'outils ou une fenêtre secondaire.

Figure 49 : Zone de texte standard



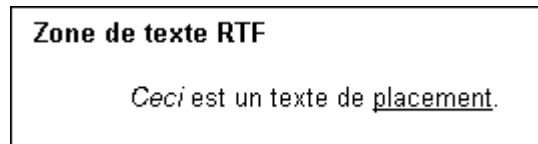
La zone de texte standard offre les fonctionnalités de base d'entrée et d'édition de texte. Les fonctionnalités d'édition comprennent l'insertion ou le retrait de caractères ainsi que le bouclage du texte (*wrapping*). Bien que les propriétés individuelles des caractères ou de paragraphes ne soient pas supportées, le contrôle entier peut supporter un réglage spécifique des caractères.

Il est également possible d'utiliser les zones de texte pour afficher du texte en mode *lecture seulement*, tout en permettant sa sélection. Lorsque vous choisissez cette option du contrôle standard, le système change automatiquement la couleur de l'arrière-plan du champ pour indiquer à l'utilisateur le changement de comportement.

Il est possible de limiter le nombre de caractères acceptés en entrée pour une zone de texte en fonction de ce qui est approprié dans un contexte donné. De plus, vous pouvez supporter la sortie automatique de la zone pour une entrée texte de longueur finie ; cela signifie que dès que le dernier caractère est entré au clavier, le centre d'attention passe au contrôle suivant. Par exemple, vous pouvez définir une sortie automatique après six caractères dans une zone définie pour contenir un code postal. Employez toutefois la sortie automatique avec modération : le changement de centre d'attention pourrait surprendre désagréablement l'utilisateur. Il est préférable de limiter son utilisation aux cas de saisies massives de données.

Zones de texte RTF (Rich Text Format)

Une zone de texte RTF, telle qu'illustrée à la figure suivante, offre les mêmes techniques de base d'édition qu'une zone de texte standard. De plus, une zone de texte RTF supporte les propriétés des caractères (polices, taille, couleur, gras, italique, etc.) ainsi que les propriétés des paragraphes (justification, indentation, numérotation, etc.). Le contrôle permet également l'impression de son contenu et l'incorporation d'objets OLE.

Figure 50 : Zone de texte RTF

Champs de texte statiques

Les champs de texte statiques (voir la figure suivante) sont généralement utilisés pour présenter des informations en mode *lecture seulement*. Ces champs de texte sont particulièrement utiles pour concevoir les étiquettes associées à des zones de texte interactives. Par exemple, si vous souhaitez que l'utilisateur entre son nom dans une zone de texte, faites précéder cette zone par un champ de texte statique où il sera écrit « Nom : ». Contrairement aux zones de texte en mode *lecture seulement*, le texte ne peut être sélectionné. Toutefois, votre application peut tout de même modifier cette zone pour refléter un changement d'état.

Figure 51 : Zone de texte statique

Mot seulement

Respect des majuscules et des minuscules

Vous pouvez également utiliser des zones de texte statique pour étiqueter d'autres contrôles ou donner de l'information à leur sujet. De plus, l'ajout d'une étiquette en texte statique permet de fournir l'accès à ces contrôles à l'aide de touches d'accès rapide. Suivez la convention usuelle pour les titres de livre, à savoir une majuscule à la première lettre du premier mot de l'étiquette ainsi que pour les noms propres. N'oubliez pas d'ajouter un deux-points (:), précédé d'un espace, à la fin du texte : cela permet d'indiquer qu'il s'agit de l'étiquette d'un contrôle.

Note : Le deux-points (:) est habituellement utilisé dans les applications Windows pour séparer une étiquette du contrôle correspondant. Néanmoins, cette façon de faire a été d'abord utilisée avec les interfaces de type caractère, qui ne bénéficient pas des possibilités de présentation visuelle qu'offrent les IPS de type graphique. Dans certains cas, l'utilisation du deux-points peut s'avérer inutile et peut même alourdir l'IPS d'une application. Lorsque c'est le cas, vous pouvez omettre le deux-points, mais seulement dans la mesure où vous pouvez le faire partout dans votre application et ce, afin de conserver une certaine cohérence pour l'utilisateur. Dans un contexte de réutilisation des composants logicielles comme celui du RSSS, il est important de suivre les conventions adoptées pour la réalisation des composants réutilisables.

Compteurs

Les compteurs sont des zones de texte qui acceptent un ensemble limité de valeurs discrètes, ensemble que l'utilisateur peut parcourir comme une boucle circulaire. Un compteur est une combinaison d'une zone de texte et d'un contrôle spécial incorporant une paire de boutons (aussi appelés contrôles *augmenter* et *diminuer*), tel qu'illustré à la figure suivante :

Figure 52 : Compteur

Lorsque l'utilisateur clique dans la zone de texte ou sur les contrôles *augmenter* ou *diminuer*, le centre d'attention se déplace dans la zone de texte du contrôle. L'utilisateur peut entrer une valeur directement au clavier ou encore utiliser les boutons pour incrémenter ou décrémenter la valeur inscrite. L'unité de changement dépend de ce que le contrôle représente : à vous de la fixer. Utilisez ces contrôles avec circonspection dans les situations où la signification des boutons pourrait être ambiguë.

7.2.4 Onglets

Les onglets forment un contrôle similaire aux divisions que l'on retrouve dans un classeur ou dans un cartable. Il sert à définir plusieurs pages logiques, ou sections d'information, à l'intérieur d'une même fenêtre.

Figure 53 : Onglets

Les étiquettes des onglets peuvent comporter du texte ou de l'information graphique. Généralement, la largeur de l'onglet s'ajuste automatiquement aux dimensions de son étiquette ; toutefois, il est possible de définir une largeur fixe pour l'ensemble des onglets. Les étiquettes doivent employer la police de caractères du système ; les règles de composition sont les mêmes que celles employées pour les menus et les boutons de commandes. Si vos étiquettes ne comportent que des graphiques, employez des info-bulles pour informer l'utilisateur sur leur signification.

Par défaut, ce contrôle affiche une seule rangée d'onglets à la fois. Bien qu'il soit possible de présenter plus d'une rangée d'onglets, il est préférable d'éviter cette approche car elle augmente la complexité de l'interface et sème parfois la confusion chez l'utilisateur : les étiquettes sont plus difficiles à lire et à atteindre. Si vous avez besoin d'un grand nombre d'onglets, séparez les pages en sous-ensembles et utilisez un autre contrôle pour passer d'un sous-ensemble à l'autre. Néanmoins, si l'action de parcourir les onglets paraît la plus appropriée, suivez les conventions indiquées dans ce guide.

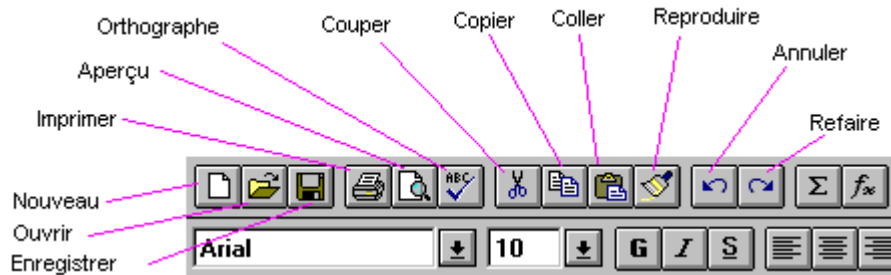
Lorsque l'utilisateur clique sur un onglet à l'aide du bouton de la souris, le centre d'attention se déplace sur cet onglet. Il est possible de se déplacer d'un onglet à l'autre à l'aide des touches FLÈCHE GAUCHE et FLÈCHE DROITE. La combinaison de touches CTRL+TAB permet aussi de se déplacer parmi les onglets. De manière optionnelle, il est également possible de définir des touches d'accès rapide dans ce but. Si l'utilisateur se déplace sur une nouvelle page à l'aide d'un onglet, le centre d'attention peut être placé sur un contrôle situé sur cette page (ex. : le bouton OK). Lorsqu'il n'y a pas de contrôle ou de champ particulier sur lequel orienter le centre d'attention, il faut le laisser sur l'onglet lui-même.

La disposition de boutons de commande (ou d'autres contrôles) à l'intérieur d'une page à onglet n'a d'effet que sur cette page. Si les boutons de commande sont placés à l'intérieur d'une fenêtre mais pas sur la page à onglet, leur effet s'applique à toute la fenêtre.

7.3 BARRES D'OUTILS ET BARRES D'ÉTAT

Comme pour les barres de menu, les barres d'outils et les barres d'état permettent de gérer un ensemble de contrôles. Une barre d'outils est un panneau qui comprend un ensemble de contrôles (voir la figure suivante) permettant l'accès rapide à des commandes ou options spécifiques. Les barres d'outils spécialisées sont parfois appelées *rubans*, *boîtes à outils* ou *palettes*.

Figure 54 : Exemple de barres d'outils



Une barre d'état (illustrée à la figure suivante) est un espace spécial, généralement situé au bas de la fenêtre, qui affiche de l'information sur l'état courant du contenu présenté dans la fenêtre. La barre d'état peut également donner de l'information contextuelle, telle que l'état du clavier, ou présenter des messages descriptifs à propos du menu sélectionné ou du bouton actif de la barre d'outils. Comme une barre d'outils, une barre d'état peut également contenir des contrôles ; toutefois, elle comprend généralement des informations non interactives ou en mode de lecture seulement.

Figure 55 : Exemple de barre d'état



8. CONCEPTION VISUELLE DES OBJETS D'INTERFACE

La perception et la compréhension du monde qui nous entoure sont influencées par ce que nous voyons. L'information visuelle est une forme de communication non verbale très puissante : elle comprend des indices qui motivent, orientent ou dérangent. Cette section aborde des principes et des recommandations de conception visuelle ou graphique à mettre en pratique dans une application pour l'environnement Windows.

Les éléments visuels de l'interface interagissent les uns avec les autres. Une conception visuelle efficace dépend fortement du contexte d'utilisation. Dans une interface de type graphique, un élément graphique et sa fonction sont entièrement interreliés. Une interface de ce type doit pouvoir s'utiliser de manière intuitive : elle doit ressembler à ce qu'elle fait et fonctionner comme son apparence le suggère.

Comme les outils de prototypage les plus répandus gèrent efficacement les attributs d'apparence des éléments d'interface (bordures, apparence de sélection, etc.), nous discuterons plutôt ici des principes et des recommandations concernant la disposition des éléments d'interface.

La taille, l'espacement et la disposition de l'information sont des facteurs critiques dans le processus de création d'un environnement cohérent, au comportement prévisible. La structure visuelle est également importante pour communiquer l'utilité des éléments affichés dans la fenêtre. En général, on doit suivre les conventions de disposition des éléments qui régissent la lecture d'information. Dans les pays occidentaux, la convention de lecture est : de gauche à droite et de haut en bas, l'information la plus importante étant généralement située au coin supérieur gauche.

8.1 POLICES ET TAILLES DES CARACTÈRES

La police par défaut du système est l'un des éléments clés de la présentation visuelle. La gestion des polices pour les éléments standard de l'interface (barres de titre, barres de menu, étiquettes des contrôles) est effectuée par les outils de développement. Toutefois, puisque l'utilisateur peut changer la police de caractères du système, il faut s'assurer de vérifier ce réglage et d'ajuster la présentation de votre interface de manière appropriée. Le système avise votre application des changements de réglage par l'entremise des fonctions **SystemParameters-Info** (polices des fenêtres principales) et **GetStockObject** (polices des fenêtres secondaires).

Le système fournit également les réglages pour la taille et les polices en fonction de nombreux facteurs comme la hauteur des barres de titres, des barres de menu et des boutons, la largeur des bordures et des barres de défilement. Prenez soin, lors de la conception de la disposition des fenêtres, de considérer ces facteurs afin que votre conception demeure à l'échelle.

8.2 MAJUSCULES

Pour afficher le texte des menus, boutons de commande, zones de texte et autres, utilisez la notation usuelle des titres, soit une seule majuscule au début du titre, à moins que le texte ne contienne un nom propre (comme le nom d'un logiciel). Par exemple :

Enregistrer
Enregistrer sous...
Rubrique d'aide Word
À propos de Microsoft Word

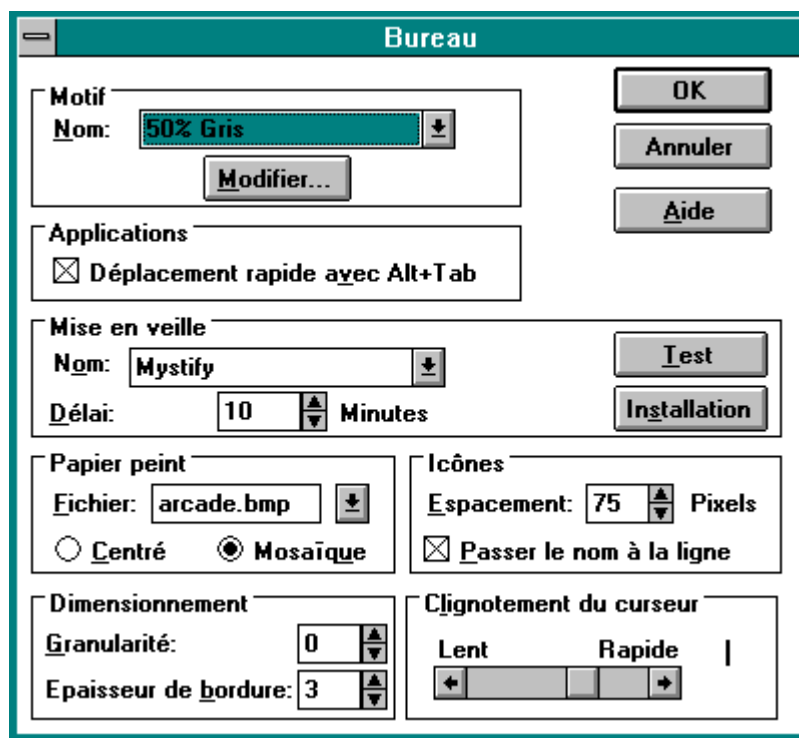
Si l'utilisateur fournit un nom pour un objet qui ne respecte pas cette convention, affichez-le tel que l'utilisateur l'a donné.

8.3 REGROUPEMENT ET ESPACEMENT

Regroupez les éléments d'interface unis par un lien logique. Vous pouvez créer des groupes à l'aide de la technique d'espacement ou encore avec des contrôles de boîte de groupe. Bien qu'il soit possible d'utiliser de la couleur pour regrouper des éléments, ce n'est pas une convention répandue et cette technique pourrait produire des effets inattendus si l'utilisateur devait changer les réglages de couleur de son système.

Conservez une marge de largeur fixe (on recommande une largeur de sept unités de boîte de dialogue) à partir de la bordure de la fenêtre. Espacez également les groupes à l'intérieur de la fenêtre. La figure suivante présente l'espacement recommandé :

Figure 56 : Disposition recommandée et espacement des contrôles et du texte



Disposez les contrôles dans la barre d'outils de manière à ce que la bordure de la barre ait au moins la même largeur que la bordure d'une fenêtre. Utilisez un minimum de quatre unités de boîte de dialogue entre chaque bouton de la barre d'outils.

8.4 ALIGNEMENT

Lorsque l'information est disposée de manière verticale, justifiez les champs à gauche. Cela peut faciliter le repérage des informations pour l'utilisateur. Les étiquettes des champs sont généralement justifiées à gauche et situées en haut ou à gauche de l'élément auquel elles s'appliquent. Lorsque les étiquettes sont placées à gauche des zones de texte, aligner la hauteur de l'étiquette avec la hauteur des caractères de la zone de texte.

8.5 DISPOSITION

Dans une fenêtre secondaire, empilez les boutons de commande principaux au coin supérieur droit ou les aligner au fond de la bordure inférieure de la fenêtre, telle qu'illustrée à la figure qui suit. S'il y a un bouton par défaut, c'est généralement le premier de l'ensemble des boutons de commande. Placer les boutons OK et *Annuler* l'un à côté de l'autre. Le dernier bouton est le bouton *Aide*, s'il existe. S'il n'y a pas de bouton OK, mais qu'il y a d'autres boutons de commande, il est préférable de placer le bouton *Annuler* après une série de boutons d'actions, mais avant le bouton *Aide*. Si un bouton de commande donné ne s'applique qu'à un groupe spécifique dans la fenêtre, il faut le placer avec ce groupe.

Figure 57 : Exemple de disposition de boutons de commande



9. CONCEPTION D'IMAGES GRAPHIQUES

Les objets graphiques sont à la base de toute interface à manipulation directe. Si certains objets ont une apparence assez standard, tels que les boutons de commande ou les listes, d'autres, tels les icônes et les curseurs, sont laissés à l'imagination des concepteurs. Pour ces derniers, il convient néanmoins de suivre des conventions de conception qui leur sont propres. Cette section traite des principaux aspects de la conception d'images graphiques.

9.1 CONCEPTION DES ICÔNES

Partout dans l'environnement Windows, les icônes servent à représenter des objets ou des tâches. Parce que le système utilise des icônes pour représenter les objets de votre logiciel, il est très important que les icônes soient non seulement efficaces, mais qu'elles puissent également communiquer leur utilité aisément.

Envisagez la conception des icônes comme un ensemble d'éléments faisant partie d'un tout cohérent, en tenant compte des relations qui les unissent entre elles et avec la tâche. Effectuez plusieurs esquisses et testez leur utilisabilité.

Tailles et types

Fournir vos icônes dans tous les formats standard : 16 x 16 pixels (16 couleurs), 32 x 32 pixels (16 couleurs) et 48 x 48 pixels (256 couleurs), telle qu'illustrées à la figure suivante. Bien qu'il soit possible d'utiliser davantage de couleurs pour les petites icônes, cela a pour conséquence d'accroître leur taille ; de plus, certaines configurations d'ordinateurs ne permettent pas l'affichage de telles icônes. Donc, si vous choisissez de fournir des icônes de petite taille à 256 couleurs, il faut s'assurer de les offrir également dans le format 16 couleurs standard.

Figure 58 : Icônes en trois tailles



Le système configure automatiquement vos icônes pour l'affichage monochrome : toutefois, il serait préférable de vérifier vous-même l'apparence monochrome de votre icône. Si le résultat n'est pas satisfaisant, il faut concevoir une icône monochrome en plus de l'icône couleur.

Enregistrez les icônes que vous fournissez dans le registre du système. Si votre logiciel ne peut enregistrer aucune icône, le système en fournira une automatiquement, basée sur l'icône de votre application. Toutefois, elle ne sera sans doute pas aussi détaillée que celle que vous pourriez concevoir vous-même.

Style des icônes

Pour faciliter le rappel, c'est-à-dire la faculté de se remémorer un objet ou un événement, il importe de concevoir des icônes simples, claires et nettes. Il faut vous assurer que les versions réduites de vos icônes soient aussi semblables que possible à l'original. En général, il est préférable de conserver la forme générale et les caractéristiques spécifiques de l'icône. Les icônes de 48 x 48 pixels peuvent être dessinées en 256 couleurs. Cela donne un effet très réaliste ; toutefois, demeurez simple et soyez prudent dans l'utilisation de la couleur. Si votre logiciel est conçu pour fonctionner sur des affichages à 256 couleurs, il faut s'assurer d'utiliser des couleurs provenant de la palette standard du système.

9.2 CONCEPTION DES POINTEURS

L'apparence d'un pointeur peut servir à aider l'utilisateur à identifier des objets et à fournir un retour d'information à propos de certaines conditions ou états. Toutefois, utilisez les changements de pointeurs avec modération afin que l'utilisateur ne soit pas distrait par des clignotements ou des changements excessifs du pointeur lorsqu'il parcourt l'écran. L'une des manières de contourner cet inconvénient consiste à attendre un court délai avant d'effectuer des changements de pointeurs qui ne sont pas critiques.

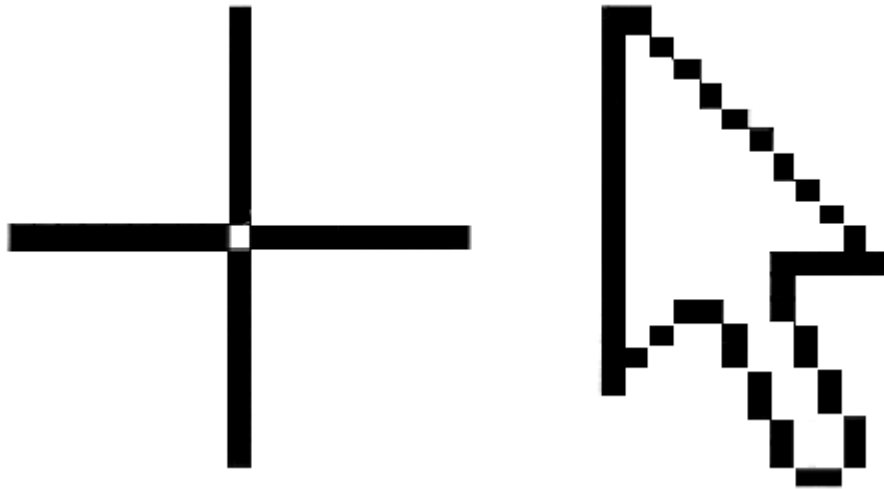
Note : Pour plus d'information sur les pointeurs, voir la section 2 – dispositifs d'interaction.

Lorsque l'apparence d'un pointeur est modifiée pour offrir un retour d'information, le changement ne doit avoir lieu que sur la zone où cet état s'applique. Par exemple, vous pouvez utiliser le sablier pour indiquer qu'une fenêtre est temporairement non interactive ; si le pointeur se déplace entre-temps sur une fenêtre interactive, changez le pointeur pour indiquer cet état. Si le traitement d'une opération rend l'interface entière non interactive, affichez le sablier là où se trouve le pointeur.

Le retour d'information offert par le pointeur ne suffit pas toujours. Par exemple, pour un traitement d'une durée de plus de quelques secondes, il est préférable d'utiliser un indicateur de progression indiquant l'évolution du traitement, le temps écoulé ou une estimation du temps qu'il reste avant la complétion de l'opération.

Employez le pointeur qui correspond le mieux au contexte de l'activité. Le pointeur en I (*I beam pointer*) est préférable pour du texte alors que le pointeur en forme de flèche usuel est utilisé pour la plupart des opérations *Glisser-Déposer* ; il est parfois modifié pour représenter les opérations *Copier* et *Lier*.

L'emplacement du point d'impact du pointeur (tel qu'illustré à la figure suivante) est important pour permettre à l'utilisateur d'atteindre sa cible. L'apparence du pointeur devrait permettre la localisation du point d'impact de manière intuitive. Par exemple, pour un pointeur en croix, il est convenu que le point d'impact se situe à l'intersection des segments.

Figure 59 : Points d'impact des pointeurs

L'animation d'un pointeur est une approche très efficace pour communiquer de l'information. Toutefois, garder en tête que l'objectif consiste à fournir un retour d'information, pas à distraire l'utilisateur. De plus, l'animation du pointeur ne doit pas restreindre l'habilité de l'utilisateur à interagir avec l'interface.

9.3 APPARENCE SÉLECTIONNÉE DES OBJETS

Lorsque l'utilisateur sélectionne un objet de l'interface, il est très important de lui présenter un retour d'information afin de confirmer que l'opération a réussi. L'une des techniques les plus répandues à cette fin consiste à changer l'apparence de l'objet sélectionné par surlignage (*Highlighting*) de l'objet ou de son arrière-plan. La figure suivante présente un exemple :

Figure 60 : Exemple d'un objet d'apparence surlignée



10. AIDE À L'UTILISATEUR

L'aide en ligne à l'utilisateur est une importante part de la conception d'un produit et peut être implantée de plusieurs manières. Le contenu de l'aide peut être de l'information contextuelle, procédurale, explicative ou didacticielle. Mais l'aide devrait toujours être simple, efficiente et pertinente, de manière à ce que l'utilisateur puisse obtenir l'information recherchée sans s'égarer dans l'interface. Cette section présente une description des approches répandues d'aide en ligne pour l'utilisateur et des recommandations pour leur mise en place. Pour plus d'information sur la rédaction de l'aide en ligne, voir la documentation incluse dans Microsoft Win32 Software Development Kit (SDK).

10.1 ACCÈS À L'AIDE

Les utilisateur ont accès à l'aide par le biais du menu ? (*Aide*), de la touche d'*Aide* (F1) ou par le mode d'*Aide* (MAJ+F1), suivi d'un cliquage de la souris sur l'élément pour lequel on désire de l'aide.

10.2 MENU ? (*AIDE*)

Le menu ? (*Aide*) est toujours le dernier menu situé à l'extrême droite de la barre de menu. Il est représenté par un point d'interrogation (?).

Figure 61 : Menu ? (Aide)



10.2.1 Contenu standard du menu ? (Aide)

Le menu ? (Aide) comprend les éléments *Index*, *Rechercher l'Aide sur...*, *Utiliser l'Aide* et *À propos de Nom de l'application*, dans l'ordre indiqué. D'autres éléments peuvent s'ajouter.

Tableau 4 : Éléments standard du menu ? (Aide)

Éléments	Fonction
<u>I</u> ndex	Ouvre la fenêtre d'aide et affiche la liste des principaux sujets en ordre alphabétique.
<u>R</u> echercher l'Aide sur...	Ouvre la fenêtre d'aide et affiche une fenêtre de dialogue qui permet à l'utilisateur de rechercher des sujets à l'aide de mots clés entrés au clavier.
<u>U</u> tiliser l'Aide	Ouvre la fenêtre d'aide et la liste des instructions pour l'utilisation de l'Aide.
<u>À</u> propos de <i>Nom de l'application</i>	Affiche une boîte de dialogue standard présentant le nom de l'application, le numéro de version, le message de droits d'auteur, l'icône, le numéro de série ainsi que le nom de l'utilisateur. La fenêtre peut également afficher de l'information additionnelle, telle que la quantité de mémoire disponible.

10.2.2 Contenu facultatif du menu ? (*Aide*)

Le menu ? (*Aide*) peut également comporter des éléments additionnels donnant accès à des catégories plus larges de l'*Aide* : par exemple *Commandes*, *Procédures*, *Clavier*. Si l'application comporte un didacticiel, le menu peut alors avoir un élément *Didacticiel* qui permet de l'activer.

10.3 TOUCHE D'AIDE

Lorsque l'utilisateur appuie sur la touche d'*Aide* (F1), la fenêtre d'aide apparaît. Lorsque cela est possible, le contenu affiché dans la fenêtre d'aide devrait être dicté par le contexte, c'est-à-dire que le contenu devrait présenter de l'information sur l'élément actuellement actif de l'interface. Par exemple, si une commande d'un menu est sélectionnée, la fenêtre d'aide devrait afficher de l'information sur cette commande. Si une boîte de message est affichée, la fenêtre d'aide devrait afficher des informations additionnelles à propos du message. S'il n'est pas possible d'offrir de l'aide contextuelle, alors la fenêtre devrait afficher la liste des sujets pour lesquels l'information est disponible.

10.4 MODE D'AIDE

Le mode d'*Aide* est une extension facultative à l'aide standard. Pour passer au mode d'*Aide*, l'utilisateur appuie sur la combinaison de touches MAJ+F1 : le pointeur passe de sa forme normale à sa forme pointeur d'*Aide* (le pointeur standard accolé à un point d'interrogation). Pour annuler le mode d'*Aide*, l'utilisateur n'a qu'à appuyer sur la touche ÉCHAP. Lorsqu'il est en mode d'*Aide*, l'utilisateur se positionne au-dessus de l'élément d'interface pour lequel il désire avoir de l'aide. Lorsque l'utilisateur clique sur le bouton de la souris, la fenêtre d'aide apparaît en affichant l'information relative à l'élément d'interface en question. Il faut toujours offrir l'aide la plus spécifique possible au contexte de travail, dans la mesure où il est possible de déterminer adéquatement ce contexte.

10.5 AIDE CONTEXTUELLE : BARRES D'ÉTAT

L'aide contextuelle fournit à l'utilisateur de l'information sur un objet donné et son contexte. Entre autres techniques, il est possible d'afficher cette information sur la barre d'état de la fenêtre.

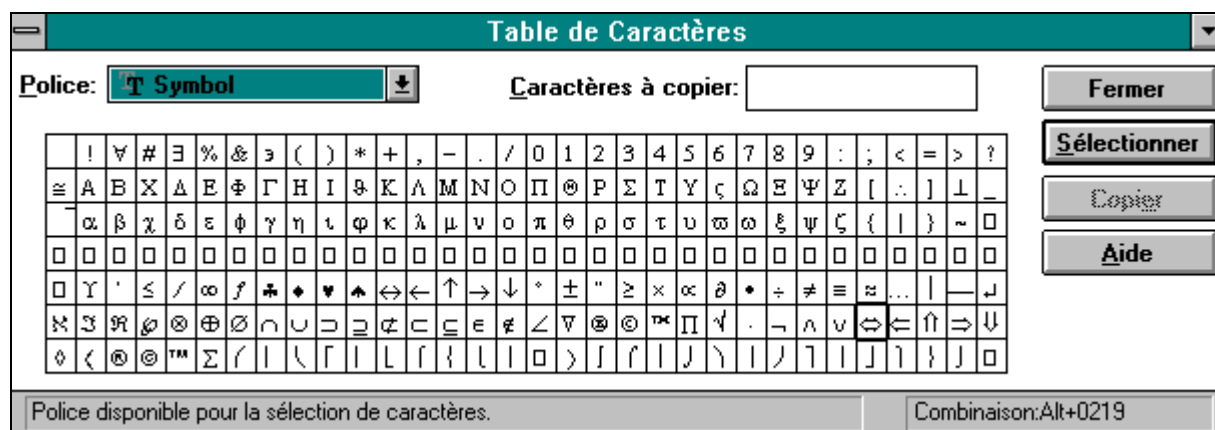
10.5.1 Messages de barres d'état

Il est possible d'utiliser la barre d'état pour offrir de l'aide contextuelle à l'utilisateur. Toutefois, si vous affichez la barre d'état dans votre application, ne pas l'employer pour afficher de l'information ou donner accès à des fonctions qui ne sont pas offertes autrement.

De plus, comme la barre d'état n'est pas nécessairement située près de l'aire de travail de l'utilisateur, ce dernier pourrait ne pas voir les messages affichés sur la barre d'état. Par conséquent, il est préférable d'utiliser les barres d'état comme une forme d'aide en ligne d'appoint.

En plus d'afficher de l'information sur l'état et le contexte de l'activité qui se déroule dans la fenêtre, il est possible d'utiliser la barre d'état pour afficher des messages descriptifs à propos des barres d'outils ou des boutons de commande, tel qu'illustré à la figure suivante. Il faut bien sûr que la fenêtre soit active pour que les messages s'affichent. Lorsque l'utilisateur déplace le pointeur sur un bouton de barre d'outils ou encore si l'utilisateur appuie sur le bouton droit de la souris alors que le pointeur est situé au-dessus d'un menu ou d'un bouton de commande, affichez un court message décrivant l'usage du contrôle en question.

Figure 62 : Message de barre d'état décrivant un élément de menu



Un message de barre d'état peut incorporer un indicateur de progression ou d'autres formes de retour d'information au sujet d'une opération en cours de traitement amorcée par l'utilisateur dans cette fenêtre, telle que l'impression ou l'enregistrement d'un fichier. Bien qu'il vous soit possible d'afficher des informations de progression dans une boîte de dialogue, l'utilisation de la barre d'état permet d'éviter l'obstruction de la fenêtre principale.

10.5.2 Recommandations pour la rédaction de messages de barres d'état

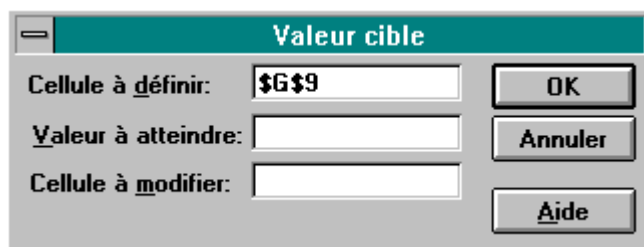
Lors de la rédaction des messages de barre d'état, commencez le message par un verbe conjugué au présent ; employez des termes familiers et évitez le jargon. Soyez aussi bref que possible afin que le texte se lise aisément mais évitez les abréviations.

Soyez constructif, pas seulement descriptif : informez l'utilisateur du but de la commande. Lorsque vous décrivez une commande ayant une fonction spécifique, employez les mots spécifiques de la commande. Si la commande s'applique à plusieurs fonctions, tentez de résumer.

Lorsque vous définissez des messages pour vos menus et barres d'outils, n'oubliez pas de concevoir des messages expliquant leur état : par exemple, la raison pour laquelle l'élément n'est pas activé.

10.6 AIDE DANS LES BOÎTES DE DIALOGUE ET DE MESSAGE

Vous pouvez présenter de l'aide contextuelle pour des feuilles de propriétés, des boîtes de dialogue ou des boîtes de message. Cette aide est accessible de deux manières : en appuyant sur la touche F1 ou, pour une approche plus visuelle, en incluant un bouton d'aide dans la fenêtre en question, tel qu'illustré à la figure suivante. Lorsque l'utilisateur choisit le bouton de commande *Aide*, affichez l'aide dans une autre fenêtre secondaire, que l'utilisateur pourra fermer lorsqu'il aura terminé de lire le message.

Figure 63 : Bouton de commande *Aide* dans une fenêtre secondaire

Le bouton de commande *Aide* doit fournir un survol, un résumé ou une explication concernant la fenêtre. Par exemple, pour une boîte de message, cette commande peut donner des explications supplémentaires sur les causes et les solutions possibles du problème. Le bouton d'aide est une forme d'aide d'appoint et ne doit pas compenser une mauvaise conception de la fenêtre secondaire.

CONCLUSION

Ce guide de style a décrit les principales composantes de l'environnement Windows 3.1 : les objets des interfaces ainsi que la gestion des interactions des utilisateurs avec ces objets. Il s'agit d'un outil destiné aux concepteurs, aux développeurs et aux autres personnes engagées dans la conception d'IPS pour le compte du RSSS. Il permet d'assurer la normalisation des environnements informatiques ainsi que la cohérence des choix de conception. Le *Guide de style* fait partie intégrante du guide de conception produit dans le cadre du projet *Guide de conception d'interactions personnes-systèmes* (IPS) du réseau de la santé et des services sociaux (RSSS).

ANNEXE 1

Collaborations

COLLABORATIONS

Le comité directeur du projet Guide de conception d'interactions personnes-systèmes du réseau sociosanitaire

Ministère de la Santé et des Services sociaux	Claude Bouchard	Service orientations, planification et normes - Groupe de soutien à la mise en œuvre des orientations technologiques (SOPN – SMOOT)
	Pierre Desbiens Lucie Dion	Service informatisation du réseau Service systèmes ministériels
Régies régionales	Daniel Beaudry Isabelle Fubini Bruno Paquet	RRSSS du Bas-St-Laurent RRSSS du Bas-St-Laurent RRSSS de Montréal-Centre
CLSC	Chantal Morand	CLSC Samuel de Champlain
Centres hospitaliers	Jacques Lareau	Centre hospitalier Pierre Boucher
CHSLD et CR	Gabriel Bélanger Marc-André Roy	Centre hospitalier Robert-Giffard Centre François-Charon

Le chef de projet

Ministère de la Santé et des Services sociaux	Bruno Asselin	SOPN – SMOOT
---	---------------	--------------

L'équipe de réalisation

Centre de recherche informatique de Montréal (CRIM)	Daniel Engelberg	Agent de recherche niveau II
	Jean-Yves Fiset	Directeur scientifique
	Denis Lacasse	Chargé de projet
	Guy Leblanc	Conseiller
	Évelyne Millien	Agent de recherche niveau II
	Alain Robillard-Bastien	Agent de recherche niveau I
	Roxanne St-Cyr	Agent de recherche niveau I
	Sylvie Turcotte	Conseiller

Soutien à la vérification et à la production du document

SMOOT	Jocelyne Gagné Manon Ouellet
-------	---------------------------------

Le comité consultatif sur la normalisation et l'expérimentation

Ministère de la Santé et des Services sociaux	Gilles Cadieux Jean Pierre Defoy	Service systèmes ministériels SOPN – SMOOT
Régies régionales	François Dépault Denis Deslauriers Robert Limane Mireille Nadeau	RRSSS de l'Estrie RRSSS du Bas-St-Laurent RRSSS de Québec et RRSSS de la Côte-Nord RRSSS du Bas-St-Laurent
Établissements	Jean-Pierre Chenel Denis Laniel Suzie Larrivée	Hôpital Notre-Dame Centre hospitalier Pierre Boucher CLSC Mercier-Est/Anjou
Firmes externes	Pierre Boutet	Groupe DMR inc.

Le groupe d'assurance-qualité

Ministère de la Santé et des Services sociaux	Sylvie Duchemin Daniel Fortin Guy Simard	SOPN – SMOOT SOPN – SMOOT Service systèmes ministériels
Régies régionales	Alain Desjardins	RRSSS de Québec