

**Master Informatique 1ère année / MIAGE - Université Paris-Sud**  
**Interfaces et Systèmes Interactifs**  
**Examen - 3 mai 2012 - 3h**

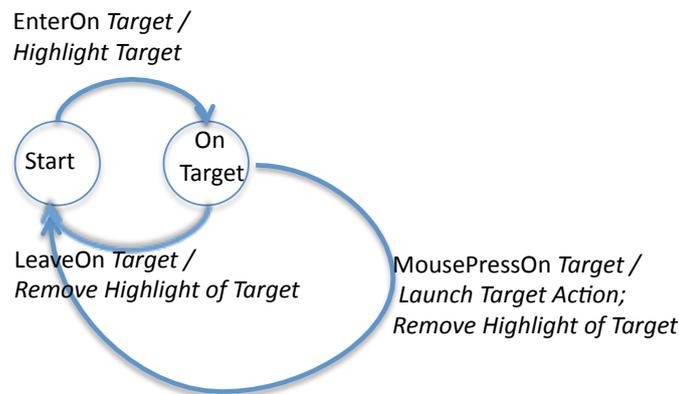
Documents autorisés : tous documents papier.  
Lisez l'énoncé en entier. Soyez clairs, précis et concis.

**A. (5 points) Modélisation de l'interaction - 3 parties**

**Note :** dans tout cet exercice on demande de définir les machines à états sous forme graphique et les actions avec du pseudo-code (pas du Java). Vous pouvez introduire des classes / fonctions / méthodes à condition de les définir avec assez de précision : soit une description informelle, soit du pseudo-code.

**Il est demandé d'être précis dans vos réponses, notamment en expliquant comment sont traités les cas limites.**

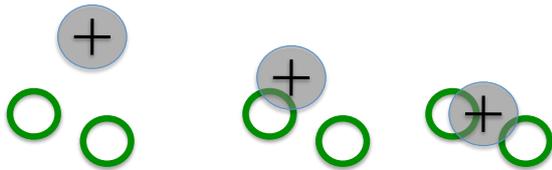
Nous allons examiner des techniques pour accélérer la sélection des cibles (icônes, boutons, etc) dans une interface sur un ordinateur de bureau (interaction avec souris). Voici une machine à état qui donne du feedback sur la cible qui est sous la souris, et qui déclenche une action en cliquant sur la cible (ouvre application, action liée au bouton, etc.) :



Donner les machines à état (une par exercice) qui permettent d'exprimer ces techniques. Pensez toujours au **feedback** de la cible qui sera sélectionnée si l'utilisateur clique.

**Rappel:** Un Etat (cercle) est un état de l'interaction, une Transition (flèche) est un événement d'entrée (Mouse Up, Down, Move, événement clavier, ...), on peut avoir des actions associées aux transitions (après le "/"), et des *guard(condition)* qui sont des conditions associées aux transitions (après le "&"). Une Transition peut avoir lieu seulement si son *guard()* est évalué à «vrai».

**(i)** « Area Cursor » : le curseur de notre souris n'est pas un point, mais un cercle. Quand le cercle de notre curseur intersecte une cible, on peut activer la cible en cliquant (*MousePress*) même si le curseur (croix) de notre souris n'est pas directement sur la cible. Le cercle de notre curseur a une taille *WIDTH* et on l'appelle «zone d'activation». Si notre zone d'activation intersecte plus d'une cible, la cible qui est la plus proche du centre de la zone d'activation est sélectionnée.



*Area Cursor : En cliquant avec la souris on n'active aucune cible dans l'image gauche, et la cible de gauche dans les deux autre images.*

Pour créer cette machine à état, vous pouvez utiliser les les fonctions :

```
List = IntersectTargets (MousePos, WIDTH)
```

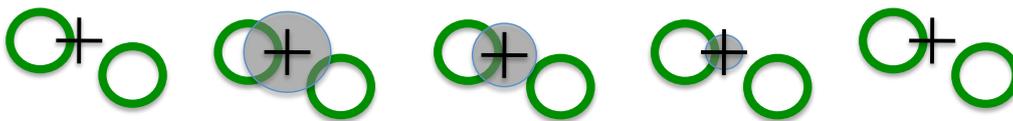
```
// Retourne la liste de cibles (Liste) qui sont intersectées par la zone d'activation par rapport à la position de souris (MousePos).
```

```
Target = CloserTarget (MousePos, List)
```

```
//Retourne la cible parmi toutes les cibles (List) qui est la plus proche de la position de notre souris (MousePos).
```

**(ii)** Le « Bubble Cursor » : lorsque l'on clique (*MousePress*) avec la souris, on déclenche systématiquement l'action liée à une cible, même si le curseur n'est pas sur une cible. La cible qu'on active est celle qui est la plus proche de la position du curseur (*MousePos*).

**(iii)** « DynaSpot » : C'est un Area Cursor qui change taille selon la vitesse de mouvement de la souris pour sélectionner les grandes cibles via un mouvement rapide, mais permet aussi de faire des sélections plus fines avec un curseur normal. Si la vitesse de mouvement de notre souris pendant un événement *MouseMove* est supérieure à une valeur *MAXSPEED*, le curseur devient un Area Cursor de taille *MAXWIDTH*. Pendant cette période de déplacement rapide le curseur agit comme un Area Cursor avec une zone d'activation *MAXWIDTH*. Si la vitesse de mouvement devient inférieure à *MAXSPEED* on commence une animation qui prend *REDUCETIME* secondes durant laquelle la taille de la zone d'activation réduit jusqu'à zéro. Après cela notre sélection des cibles est la sélection simple. Pendant la période *REDUCETIME* l'utilisateur peut changer d'avis et accélérer de nouveau jusqu'à *MAXSPEED* et re-activer l' Area Cursor.



*DynaSpot : Quand la vitesse est inférieure à MAXSPEED le curseur est normal (1ère image). Si la vitesse est supérieure à MAXSPEED c'est un Area Cursor de zone d'activation MAXWIDTH (2ème image). Si la vitesse devient inférieure à MAXSPEED la zone d'activation réduit jusqu'à devenir nulle.*

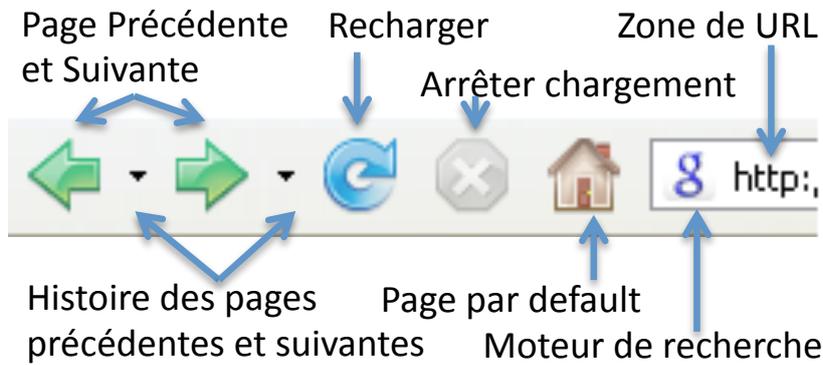
**Note :** La vitesse du curseur est définie par la déplacement du curseur *MousePos* pendant 1 sec.

Vous disposez d'une fonction *GetTime()* qui retourne le temps écoulé en millisecondes depuis le dernier appel à *ResetTime()*.

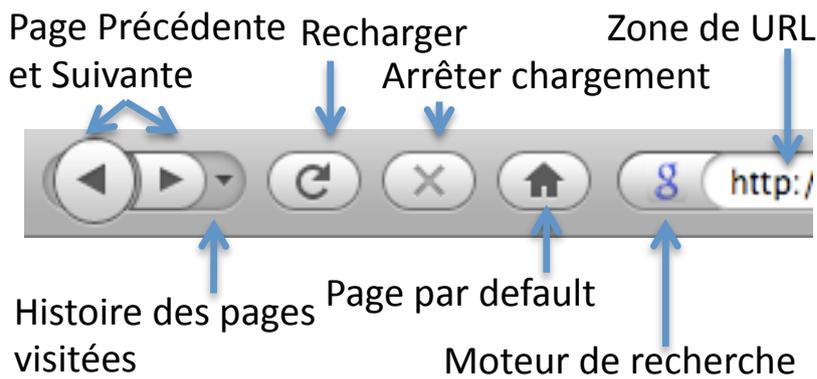
Finalement, vous avez un événement "TimeOut" dont on peut programmer le déclenchement au bout de *n* millisecondes par la fonction *Arm(n)*. Vous pouvez utiliser la méthode *Disarm()* qui annule l'appel à *Arm()* si l'événement "TimeOut" n'a pas encore été émis, et qui ne fait rien sinon.

**B. (5 points) Psychologie : Perception, cognition et mouvement & Usabilité : Catégories - 2 parties**

Une entreprise qui développe des navigateurs Web vous demande de recommander l'un des deux modèles d'interface pour leur navigateur Web (Figure b.1 et b.2). Vous allez critiquer chaque solution proposée et les bénéfices associés.



(b.1)



(b.2)

**(i)** Quels sont les avantages et les désavantages de chaque approche ? Justifiez votre critique en vous basant :

**a.** sur les aspects suivants de la performance humaine :

- i.** Performance motrice : le loi de **Fitts**.
- ii.** Perception visuelle : les lois gestaltistes de **similitude**, de **proximité** et **clôture**.
- iii.** Cognition : **attention**.

**b.** sur les catégorie d'usabilité de **guidage**

**(ii)** Quelle solution recommanderez vous à l'entreprise ?

### **C. (5 points) Modèle conceptuel - 4 parties**

Nous avons un système de tableau blanc interactif qui agit comme un tableau blanc ordinaire, c.à.d. qu'on peut écrire avec des stylos appropriés. De plus:

- chaque utilisateur a son propre stylo qu'on peut identifier (c.à.d. qu'on sait qui écrit sur le tableau),
- le tableau enregistre automatiquement dans un fichier tout son contenu s'il n'y a pas d'entrée utilisateur pendant plus de 5 min,
- les utilisateurs peuvent également sauvegarder explicitement le contenu du tableau dans un fichier et donner un nom au fichier,
- les utilisateurs peuvent accéder aux contenus déjà enregistrés (automatiquement ou explicitement). Le contenu à accéder apparaît comme un boîte de dialogue avec une liste des fichiers sauvegardés et une fonction de recherche,
- la liste de fichiers/contenus sauvegardés contient des métadonnées associées (date et heure de sauvegarde, taille du fichier, personne qui a écrit ou sauvegardé), et indique si le fichier est un auto-save ou un enregistrement déclenché par l'utilisateur.

**(i)** Proposer une façon de déclencher explicitement l'action d'enregistrement et l'accès au contenu du tableau (décrivez l'interface et interaction). Illustrez si nécessaire les interactions par des dessins.

**(ii)** Identifier les objets et les opérations du modèle conceptuel de votre interface, en justifiant vos choix.

**(iii)** Compléter la table des objets et la table des opérations (Table Fonctionnelle) décrivant le modèle conceptuel de cette interface. Illustrez si nécessaire les interactions par des dessins.

Objets	Représentations	Propriétés	Opérations
Contenu Tableau	Texte and dessines sur le tableau	...	...
Fichiers sauvegardés	...	...	
....			

Opérations	Commandes/Interaction	Feedback	Réponses
Sauvegarder	...	...	Contenu du tableau sauvegardé
...			

**(Reproduisez ces tableaux sur votre copie)**

**(iv)** Dans votre concept, comment est-ce que vos utilisateurs font la saisie de texte avec leur stylos pour nommer les fichiers?

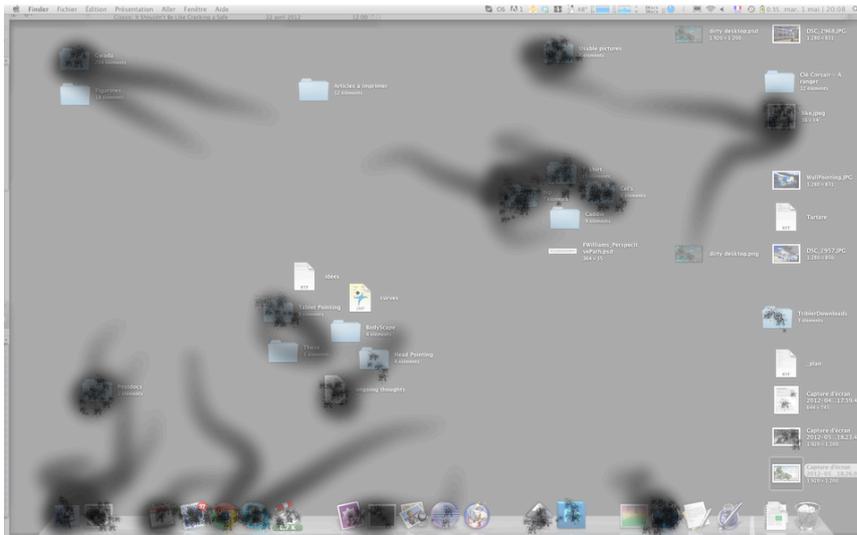
**a.** Critiquez votre choix : Donnez au minimum un avantage et un désavantage de votre choix en utilisant des concepts vus en classe (ex catégories d'usabilité, loi Fitts, principes de conception, cognition, etc).

**b.** Proposez une alternative de saisie de texte pour votre système et critiquez la de la même façon (un avantage et un désavantage).

## D. (5 points) - Modèle MVC - 2 parties

Pointer (c'est-à-dire déplacer un curseur au dessus d'une cible) est une tâche extrêmement courante, que ce soit lors d'une utilisation personnelle ou professionnelle d'un ordinateur. De nombreux travaux de recherche ont eu pour but d'améliorer les performances de pointage, c'est-à-dire réduire le temps nécessaire pour acquérir la cible et d'en augmenter la précision.

La technique suivante consiste à enregistrer en continu les positions du curseur sur le bureau virtuel ou dans une application. La fréquence d'affichage du curseur étant fixe, une zone dans laquelle ont été enregistrées beaucoup de positions du curseur est une zone dans laquelle le curseur est resté longtemps et s'est déplacé lentement, donc une zone dans laquelle il est probable que se situent des cibles fréquentes ou difficiles à pointer. A l'inverse, une zone avec peu de positions enregistrées signifie que l'utilisateur y est peu passé, ou s'y est déplacé rapidement. Avec cette technique, le CD gain<sup>1</sup> du curseur est ajusté dynamiquement en fonction de la densité de "visites" précédentes du curseur sur sa position courante : le gain sera élevé dans une zone à faible densité (mouvements rapides) et bas dans une zone à densité élevée (mouvements précis).



Exemple d'utilisation de notre technique sur un bureau virtuel. Sont représentées en noir les zones dans lesquelles le curseur est passé le plus souvent.

Il s'agit d'une technique évolutive : lors de sa première utilisation, aucun point n'a été enregistré et toutes les zones ont la même densité, donc le même CD gain (1). C'est au cours de l'utilisation que les positions du curseur vont être enregistrées et que la "carte" des densités va devenir de plus en plus précise.

**(i)** Comment implémenter cette technique ? Présentez un schéma expliqué du fonctionnement de la technique à l'aide du modèle MVC. (*suggestion* : identifiez les points principaux de l'algorithme et de la technique et expliquez quelle partie du MVC les gère)

**(ii)** Voyez-vous des problèmes à cette technique ? Si oui, lesquels, et voyez-vous des solutions ?

---

<sup>1</sup> Multiplicateur appliqué au déplacement physique de la souris. Un multiplicateur élevé cause des déplacements rapides et inversement. Typiquement, un CD gain supérieur à 1 représente des mouvements de curseur plus rapides que le mouvement de la souris, et inversement.