

## La saisie des caractères spéciaux

Le système d'exploitation permet de choisir parmi un ensemble prédéfini de claviers internationaux (*Démarrer, Panneau de configuration, Clavier...*). Mais il existe des utilitaires qui offre en plus la possibilité de créer son propre clavier et de le rendre accessible au système, donc à tous les logiciels tournant sur le système (Word, Excel, Shoebox...).

### Microsoft Keyboard Layout Creator<sup>1</sup>

Ses spécificités sont :

- Chargement d'un clavier système existant
- Représentation graphique du clavier
- Choix d'une police
- Affectation des caractères d'une police aux touches du clavier : minuscule, majuscule (Maj), contrôle (Ctrl), Alt/Ctrl, Maj/Alt/Ctrl. par
  - Alt/0xxx pour le jeu CP1252,
  - Couper-coller ou glisser-déposer à parti d'une charte, ou fichier Word.
  - Unicode: U+xxxx, U+xxxx[xx], ou \xxxx
- Définition de *touches mortes*
- Test du clavier
- Enregistrement du fichier source (klc)
- Compilation d'un fichier d'installation (.msi) servant à installer ou désinstaller la bibliothèque Layoutxx.dll dans Windows/system32
- Installation d'un clavier: Panneau de configuration, Clavier, Ajouter un clavier...

### Evaluation

MsKLC est un outil gratuit qui permet de configurer simplement des claviers personnalisés, à partir de Windows 2000. L'utilisation la plus simple consiste à affecter directement à des touches du clavier les caractères difficilement accessibles. Pour une police phonétique par exemple, on pourra ainsi remplacer les majuscules latines par des caractères spéciaux de l'API. On pourra aussi utiliser des combinaisons avec les touches Ctrl, Alt/Ctrl et Maj/Alt/Ctrl, mais il faudra veiller à ce que ces combinaisons n'interfèrent pas avec celles utilisées par les logiciels ou par le système lui-même pour accéder directement à des fonctions (Ctrl/a, Ctrl/c, Ctrl/v...).

La technique de la *touche morte*, couramment utilisée pour accéder à l'accent circonflexe et au tréma sur les claviers français est extrapolée par ce logiciel qui permet de sélectionner d'autres touches rarement utilisées et bien positionnées sur le clavier pour servir à cet effet.

Une fois qu'une touche est définie comme touche morte, il est possible de définir les codes que doit renvoyer le clavier pour les différentes combinaisons de cette touche morte avec d'autres touches.

### Accès à des caractères *spéciaux*

On pourra par exemple définir la touche "<", en bas à gauche du clavier, comme touche morte pour accéder à des caractères spéciaux d'une police, comme "epsilon" en combinaison avec la touche "e", "o ouvert" avec "o", "n vélaire" avec "n", etc. Cette touche "<" fonctionne alors comme la touche "tréma" : elle n'affiche rien, mais attend la touche suivante pour renvoyer le caractère voulu.

<	e	o	a	n
	ε	ɔ	ɑ	ŋ

<sup>1</sup> L'utilitaire **msklc.exe** (<http://www.microsoft.com/globaldev/tools/msklc.msp>) nécessite l'installation préalable du paquetage *.NET Framework* (**dotnetfx.exe** à <http://www.microsoft.com/downloads/>)

## Diacritiques

Comme exemple de diacritique, on pourra choisir de définir la touche "\_" pour rajouter un de longueur aux voyelles. A ce niveau deux cas peuvent se présenter suivant le type de police utilisée :

– soit ce trait est représenté par un caractère diacritique flottant, de largeur nulle qui vient se positionner sur la lettre précédente ; il suffira d'affecter à la touche "\_" le code du caractère spécial "macron" (U+0304 dans une police Unicode).

a_	e_	ε_
ā	ē	ē

– soit ce trait est intégré au caractère de base sous la forme d'un caractère composite unique. Dans ce cas, on définira la touche "\_" comme *touche morte* et on définira les combinaisons de celle-ci avec les autres touches concernées par la longueur (voyelles). On remarquera que toutes les voyelles n'existent pas précomposées avec toutes les diacritiques et on privilégiera de ce fait l'approche des diacritiques flottantes.

_	a	e	o
	ā	ē	ō

## Application

- Télécharger Microsoft Keyboard Layout Creator (fichier *msklc.exe* à <http://www.microsoft.com/globaldev/tools/msklc.msp>)
- Cliquer sur *msklc.exe* pour installer Microsoft Keyboard Layout Creator.

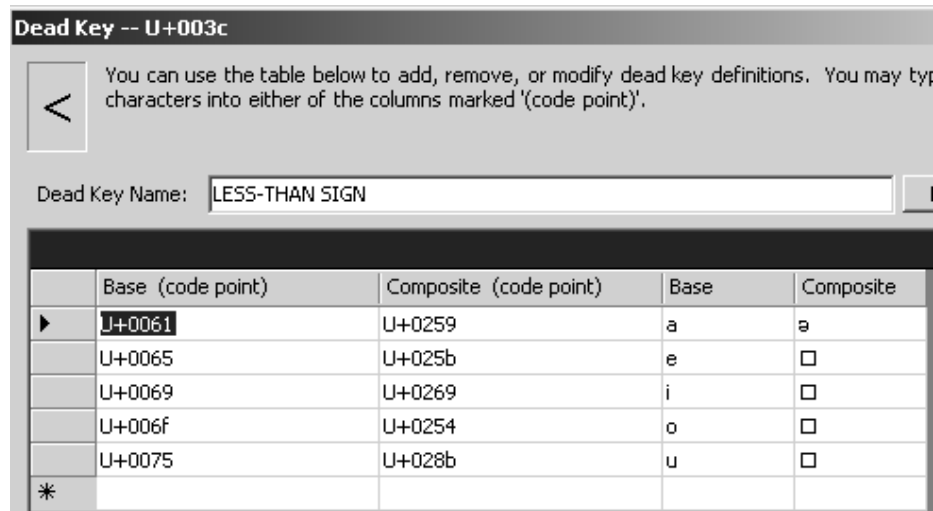
[ Si nécessaire, télécharger .NET Framework Version 1.1 Redistributable Package (fichier *dotnetfx.exe*) et exécuter-le, puis relancer l'installation de MSKLC ]

## Définir une touche morte

- Cliquer sur les boutons "..." pour choisir les dossiers de sauvegarde du clavier
- *View, Options*, Choisir le type de clavier
- Importer un clavier : *File, Load existing keyboard, Français*
- Cliquer sur la touche à définir : < par exemple
- Cliquer sur *All* pour faire apparaître la fenêtre complète
- Cocher la case Dead Key View
- Cocher sur la case DeadKey à droite du symbole "<"
- Cliquer sur le bouton "..."

Il s'agit maintenant de définir pour chaque touche devant entrer en combinaison avec la touche morte "<" le caractère correspondant. Par exemple pour obtenir un "ä" sur la touche "a" en combinaison avec la touche morte "<", on fera ce qui suit :

- Taper la touche "a" dans la cellule <base>
- Dans la cellule correspondante <composite> mettez le code Unicode du caractère "ä", soit "U+0041". Le caractère "ä" devrait apparaître dans la cellule <composite> à droite.
- Faire de même pour les autres voyelles concernées (e, i, o, u par exemple pour avoir respectivement é, î, ô, u).
- Puis cliquer sur OK



## Vérifier le fonctionnement du clavier

- Projet, Test Keyboard Layout. Une fenêtre s'ouvre où les combinaisons peuvent être testées.

## Créer le clavier

- Projet, Propriétés.
  - Name: nom du fichier .dll qui sera créé dans le dossier
  - Language: Choisir la langue sur laquelle s'appuie ce clavier. En choisissant *French*, l'icône "FR" apparaîtra lors de la sélection du clavier
  - **Description**: nom sous lequel ce clavier figurera dans la liste des claviers disponibles
  - Projet, Buid Dll and Setup Package. Un dossier est créé contenant un fichier .msi

## Installer le clavier dans Windows

- Aller dans le dossier où a été enregistré le clavier (voir *View, Options*)
- Double-cliquer sur le fichier .msi
- Démarrer, Panneau de configuration, Clavier, Paramètres régionaux d'entrée. Ajouter.
  - Choisir comme paramètres régionaux d'entrée le clavier ayant servi de base à la modification (en général, Français (France))
  - Choisir comme *configuration clavier/IME* le clavier généré (voir **Description**) et faire OK

## Utiliser le clavier

Pour choisir le clavier modifié, cliquer sur l'icône "FR" en bas à droite de l'écran et choisir le clavier voulu.

## Exercice: Définir les touches mortes pour des diacritiques

Soit à créer un clavier pour la translittération du chinois en pinyin. Quatre tons sont à définir.

Touche	voyelle						
	N° ton	a	e	i	o	u	ü <sup>2</sup>
,	1er <sup>ˊ</sup>	ā 257	ē 275	ī 299	ō 333	ū 363	ǖ 470
;	2e <sup>ˊˊ</sup>	á 225	é 233	í 237	ó 243	ú 250	ǘ 472
:	3e <sup>ˇ</sup>	ǎ 462	ě 283	ǐ 464	ǒ 466	ǔ 468	ǚ 474
!	4e <sup>ˋ</sup>	à 224	è 232	ì 236	ò 242	ù 249	ǜ 476

<sup>2</sup> Pour un accès direct au caractère "ü", on pourra commencer par redéfinir la touche "\*" en "ü" (et la touche "µ" en "\*"). Puis on définira le comportement des touches mortes avec la touche "\*" en guise de caractère "ü".

Ses atouts :

- Langage élaboré de script définissant le comportement du clavier
- Choix d'une police pour afficher les caractères spéciaux
- Accès aux caractères spéciaux par code décimal (police 8 bits) ou Unicode (U+xxxx)
- Affectation d'une séquence de caractères à une touche
- Définition de groupe de caractères permettant l'application simplifiée de règles communes aux caractères du groupe
- Notion de contexte permettant le remplacement du dernier caractère affiché en fonction de la touche tapée
- Définition de *touches mortes* pouvant se *casca*der
- Test/débogage du clavier
- Enregistrement du fichier source (kmn)
- Compilation du clavier (kmx), création d'un fichier exécutable pour la distribution du *runtime* Keyman et du/des clavier(s) avec possibilité d'y inclure les polices spéciales qui s'installeront automatiquement
- Chargement du/des clavier(s) dans Keyman lancé en tâche de fond et activation d'un clavier par combinaisons de touche ou par la souris

### Evaluation

Keyman présente par rapport au logiciel de Microsoft, l'avantage de prendre en compte le *contexte* dans lequel une touche est tapée. Le dernier caractère sorti à l'écran (contexte) peut ainsi être remplacé par un autre, suivant les règles définies dans le script du clavier.

Ainsi le script suivant :

```
"e" + "<" > "ε"
```

```
"ε" + "!" > "ẽ"
```

demande que la saisie du caractère *modificateur* "<" à la suite d'un "e" provoque le remplacement du "e" par le caractère "ε". Si ce dernier est lui-même suivi du caractère "!", il sera remplacé par le caractère composite "ẽ", ce que ne permettait pas le logiciel de Microsoft.

Il est aussi possible de définir comme précédemment la touche morte "<" comme suit :

```
+ "<" > dk(1)
```

Puis de donner la règle :

```
dk(1) + "e" > "ε"
```

afin que la combinaison de touche "<e" donne le caractère voulu "ε".

La définition de l'autre touche morte :

```
+ "!" > dk(2)
```

en combinaison avec la règle :

```
dk(2) + "e" > "ẽ"
```

permet d'afficher "ẽ" par la combinaison "!e" ; elle permet de plus avec la règle suivante :

```
dk(2) dk(1) + "e" > "ẽ"
```

d'accéder au caractère "ẽ" en tapant la combinaison "<!e".

*Remarque* : La solution du caractère *modificateur* présente l'avantage que le remplacement des caractères se fait de façon explicite à l'écran au fur et à mesure que les touches sont tapées, alors que

---

<sup>3</sup> www.tavultesoft.com

les touches mortes n'affichent rien tant que le caractère de base n'est pas tapé. D'autre part, cet ordre de saisie est plus proche de l'ordre naturel de l'écriture où l'on pose le caractère de base puis les diacritiques. Une solution intermédiaire est présentée en annexe.

Le langage de script de Keyman permet d'écrire des règles synthétiques concernant un ensemble de caractères. Par exemple, la définition préalable de l'ensemble des voyelles de base et de leurs correspondantes phonétiques :

**store (voyelles) "aeiou"**

**store(voymod) "əɛɪʊ"**

permet en une règle de définir la règle d'accès aux voyelles spéciales :

**any(voyelles) + "<" > index(voymod,1)**

qui dit que la frappe de la touche "<" à la suite d'une voyelle *normale* (de la classe "voyelles") doit entraîner le remplacement de cette dernière par la voyelle spéciale correspondante (de la classe "voymod").

## Installation de Keyman

Keyman se décline en deux modules : le développeur qui sert à créer un clavier et le runtime qui sert à utiliser les claviers keyman. Le développeur est payant (version démo un mois), mais il existe une version gratuite du runtime pour une utilisation à titre privée.

Un fichier développé avec *Keyman Developer* est un fichier texte (.kmn) qui devra être compilé. C'est ce fichier compilé (.kmx) qui sera installé dans le runtime Keyman.

### Runtime

- Se connecter à <http://www.tavultesoft.com/keymandev/downloads/>
- Sélectionner **Keyman 6.2 Free Home Use Edition**
- Double-Cliquer sur le fichier pour l'installer. Une icône apparaît en bas à droite de l'écran sous la forme d'un losange gris avec un K inscrit à l'intérieur
- Clic-droit sur cette icône

## Installation d'un clavier

- Sélectionner *Keyman Configuration*
- Cliquer sur le bouton *Install Keyboard*
- Rechercher le dossier contenant le clavier à installer (ex: AO.kmx)

## Utilisation de Keyman

- Pour activer un clavier, cliquer sur l'icône de Keyman et sélectionner son logo
- Pour activer un clavier, cliquer sur l'icône de Keyman et sélectionner *No Keyman Keyboard*

## Exemple de clavier Keyman : AFRO.kmn

c Clavier pour les langues utilisant les diacritiques

c tons haut, bas, haut-bas, point souscrit, ainsi que les nasales correspondantes

c à utiliser avec une police Unicode

c Auteur Ch Chanard (chanard@vjf.cnrs.fr) 29/10/04

c LLACAN-CNRS (<http://llacan.cnrs-bellevue.fr>)

c Les diacritiques sont tapés avant le caractère de base grâce à des touches mortes.

c Les tons bas, moyen et haut sont respectivement sur les touches

c " , " ; " : " du clavier français

c le ton haut-bas est sur l'accent circonflexe

c Les tons sur voyelles nasales sont sur les majuscules des touches mortes servant aux tons seuls.

c Les tons sur voyelles nasales bas, haut-bas et haut sont respectivement sur les touches

c " ? " . " / "  
 c La nasalité seule se trouve sur " ! "  
 c ton haut-bas se tape " : " suivi de " , " puis voyelle  
 c ton bas-haut se tape : " , " suivi de " : " puis voyelle  
 c nasale ton haut-bas se tape "/" suivi de " ; " puis voyelle ou plus simplement en tapant "\$" suivi de la voyelle  
 c nasale ton bas-haut se tape "?" suivi de " : " puis voyelle

c La touche "<" sert à modifier les caractères modifiables (e,o,d, E, O, D)  
 c Le principe pour les voyelles "spéciales" portant des diacritiques,  
 c est de porter les tons sur les voyelles normales puis de les modifier avec la touche "<"  
 c L'ordre des diacritiques dans le fichier est nasale avant tons  
 c Les touches mortes suivies d'un espace affichent le caractère normal  
 c Les touches mortes suivies d'un signe "=" affichent un ton flottant

c le point souscrit est accessible par l'astérisque

```
Name "AFRO"
BITMAP "AFRO.bmp"
VERSION 6.0
begin Unicode > use(Main)
store(&mnemoniclayout) "1"
```

```
store (voyelle) "a" "e" "i" "o" "u" \
                "A" "E" "I" "O" "U"
store(voyaigu) "á" "é" "í" "ó" "ú" \
               "Á" "É" "Í" "Ó" "Ú"
store(voygrave) "à" "è" "ì" "ò" "ù" \
                "À" "È" "Ì" "Ò" "Ù"
store(voycirc) "â" "ê" "î" "ô" "û" \
               "Â" "Ê" "Î" "Ô" "Û"
store(voytilde) "ã" "ê" "ï" "ô" "û" \
                "Ã" "Ê" "Ï" "Ô" "Û"
```

```
store(voymod) U+0259 U+025B U+0269 U+0254 U+028B \
              U+018F U+0190 U+0196 U+0186 U+01B2
```

```
store (modifiable) "a" "e" "h" "i" "o" "u" \
                   "b" "c" "d" "g" "j" "k" "l" "n" \
                   "p" "q" "r" "s" "t" "y" "z" "x" \
                   "A" "E" "I" "O" "U" \
                   "B" "C" "D" "G" "J" "K" "L" "N" \
                   "P" "Q" "R" "S" "T" "Y" "Z"
```

```
store(modifie1) U+0259 U+025B U+0266 U+0269 U+0254 U+028B \
                U+0253 U+0188 U+0257 U+0260 U+0272 U+0199 "I" U+014B \
                U+01A5 U+02BC U+027E U+0283 U+01AD U+01B4 U+0292 U+0263 \
                U+018E U+0190 U+0196 U+0186 U+01B2 \
                U+0181 U+0187 U+018A U+0193 U+019D U+0198 "I" U+014A \
                U+01A4 U+02BC U+027E U+0283 U+01AC U+01B3 U+0292
```

```
store(modifie2) U+0251 "e" "h" U+0268 "o" U+0289 \
                "b" "c" U+0256 U+0263 U+025F "k" "l" "n" \
                "p" U+0294 U+027D U+0161 U+0288 "y" U+017E "x" \
                "A" "E" U+0197 "O" U+F218 \
                "B" "C" "D" U+0194 "J" "K" "L" "N" \
                "P" U+02C0 "R" U+0160 "T" "Y" U+017D
```

```
store(pointable) "a" "b" "d" "g" "h" "i" "l" "r" "s" "t" "u" "z" "q" \
                 "A" "B" "D" "G" "H" "I" "L" "R" "S" "T" "U" "Z"
```

```
store(pointe) U+0103 U+1E05 U+1E0D U+0121 U+1E25 U+0268 U+1E37 U+F213 U+1E63 U+1E6D U+0289 \
              U+1E93 U+0295 \
              U+0102 U+1E04 U+1E0C U+0120 U+1E24 U+0197 U+1E36 U+F214 U+1E62 U+1E6C \
              U+F218 U+1E92
```

```
group(Main) using keys
+ " ; " > dk(1) c bas
c + " ; " > dk(2) c moyen (ex bas-haut)
+ " : " > dk(3) c haut
+ " ! " > dk(4) c tilde
```

+ "?" > dk(5) c tilde bas  
 + "." > dk(6) c tilde bas-haut  
 + "/" > dk(7) c tilde haut  
 + "\$" > dk(HB) c haut-bas (circ)  
 + "§" > dk(8) c tilde haut-bas  
 c + "\$" > dk(9) c point souscrit  
 + ";" > dk(9) c point souscrit

any(modifiable) + "<" > index(modifie1,1)  
 any(modifiable) + ">" > index(modifie2,1)

c point souscrit (tamacek)  
 dk(9) + any(pointable) > index(pointe,2)  
 dk(9) + "A" > U+0102  
 c dk(9) + "r" > "r" U+0335  
 c dk(9) + "R" > U+F214 c "R" U+0335

c ton bas-haut  
 dk(1) dk(3) + any(voyelle) > index(voyelle,3) U+030C  
 dk(5) dk(3) + any(voyelle) > index(voyelle,3) U+0330 U+030C  
 any(voyelle) U+030C + "<" > index(voymod,1) U+030C

c ton haut-bas  
 dk(HB) + any(voyelle) > index(voyelle,2) U+0302  
 dk(3) dk(1) + any(voyelle) > index(voyelle,3) U+0302  
 dk(7) dk(1) + any(voyelle) > index(voyelle,3) U+0330 U+0302  
 any(voyelle) U+0302 + "<" > index(voymod,1) U+0302

any(voycirc) + "<" > index(voymod,1) U+0302

c ton bas  
 dk(1) + any(voyelle) > index(voyelle,2) U+0300  
 any(voyelle) U+0300 + "<" > index(voymod,1) U+0300  
 c dk(1) + any(voyelle) > index(voygrave,2)  
 c any(voygrave) + "<" > index(voymod,1) U+0300

c ton moyen (ex bas-haut)  
 c dk(2) + any(voyelle) > index(voyelle,2) U+030C  
 dk(2) + any(voyelle) > index(voyelle,2) U+0304  
 any(voyelle) U+0304 + "<" > index(voymod,1) U+0304

c ton haut  
 dk(3) + any(voyelle) > index(voyelle,2) U+0301  
 any(voyelle) U+0301 + "<" > index(voymod,1) U+0301  
 c dk(3) + any(voyelle) > index(voyaigu,2)  
 c any(voyaigu) + "<" > index(voymod,1) U+0301

c tilde  
 dk(4) + any(voyelle) > index(voyelle,2) U+0330  
 any(voyelle) U+0330 + "<" > index(voymod,1) U+0330

c tilde-bas  
 dk(5) + any(voyelle) > index(voyelle,2) U+0330 U+0300  
 any(voyelle) U+0330 U+0300 + "<" > index(voymod,1) U+0330 U+0300

c tilde bas-haut  
 dk(6) + any(voyelle) > index(voyelle,2) U+0330 U+030C c U+030C  
 any(voyelle) U+0330 U+030C + "<" > index(voymod,1) U+0330 U+030C c U+030C

c tilde-haut  
 dk(7) + any(voyelle) > index(voyelle,2) U+0330 U+0301  
 any(voyelle) U+0330 U+0301 + "<" > index(voymod,1) U+0330 U+0301

c tilde-haut-bas  
 dk(4) + any(voycirc) > index(voyelle,2) U+0330 U+0302  
 any(voyelle) U+0330 U+0302 + "<" > index(voymod,1) U+0330 U+0302

dk(8) + any(voyelle) > index(voyelle,2) U+0330 U+0302  
 any(voyelle) U+0330 U+0302 + "<" > index(voymod,1) U+0330 U+0302

dk(4) + "n" > "ñ"  
 dk(4) + "N" > "Ñ"

c tons flottants en fin de mot  
 dk(1) + "=" > " " U+0300 " "  
 dk(2) + "=" > " " U+030C " "  
 dk(3) + "=" > " " U+0301 " "  
 "¨" + "=" > " " U+0302 " "

dk(1) + " " > "; "  
 dk(2) + " " > ", "  
 dk(3) + " " > ". "  
 dk(4) + " " > "! "  
 dk(5) + " " > "? "  
 dk(6) + " " > ": "  
 dk(7) + " " > "/ "  
 dk(9) + " " > "; "

## Principe de fonctionnement du clavier AFRO

touches sensibles	b	c	d	g	j	k	l	n	p	q	r	s	t	y	z	a	e	i	o	u
modificateur <	Ḇ	Ĉ	Ḍ	Ĝ	Ĵ	Ķ		Ŋ	Ṕ	’	Ṛ	Ṣ	Ṧ	Ẏ	Ẓ	ə	ɛ	ɪ	ɔ	ʊ
modificateur >			Ḧ	ȳ	ȝ					ʔ	ṛ	ṣ			ẏ			ï		ʘ
touche morte ;	ḃ		ḏ	ḡ			!			ʔ	ṙ	ṣ	ṧ		ẏ	ă				

touches sensibles	B	C	D	G	J	K	L	N	P	R	S	T	Y	Z	A	E	I	O	U
modificateur <	Ḃ	Ĉ	Ḍ	Ĝ	Ĵ	Ķ		Ŋ	Ṕ			T	Y		ə	ɛ	ɪ	ɔ	ʊ
modificateur >											š			ž			ï		
touche morte ;	Ḃ		Ḍ	Ĝ			!			Ṛ	Ṣ	Ṧ		ẏ					

Diacritiques sur voyelles	touches mortes	!	,	?	:	/
	ex. avec o	õ	ò	ö	ó	õ