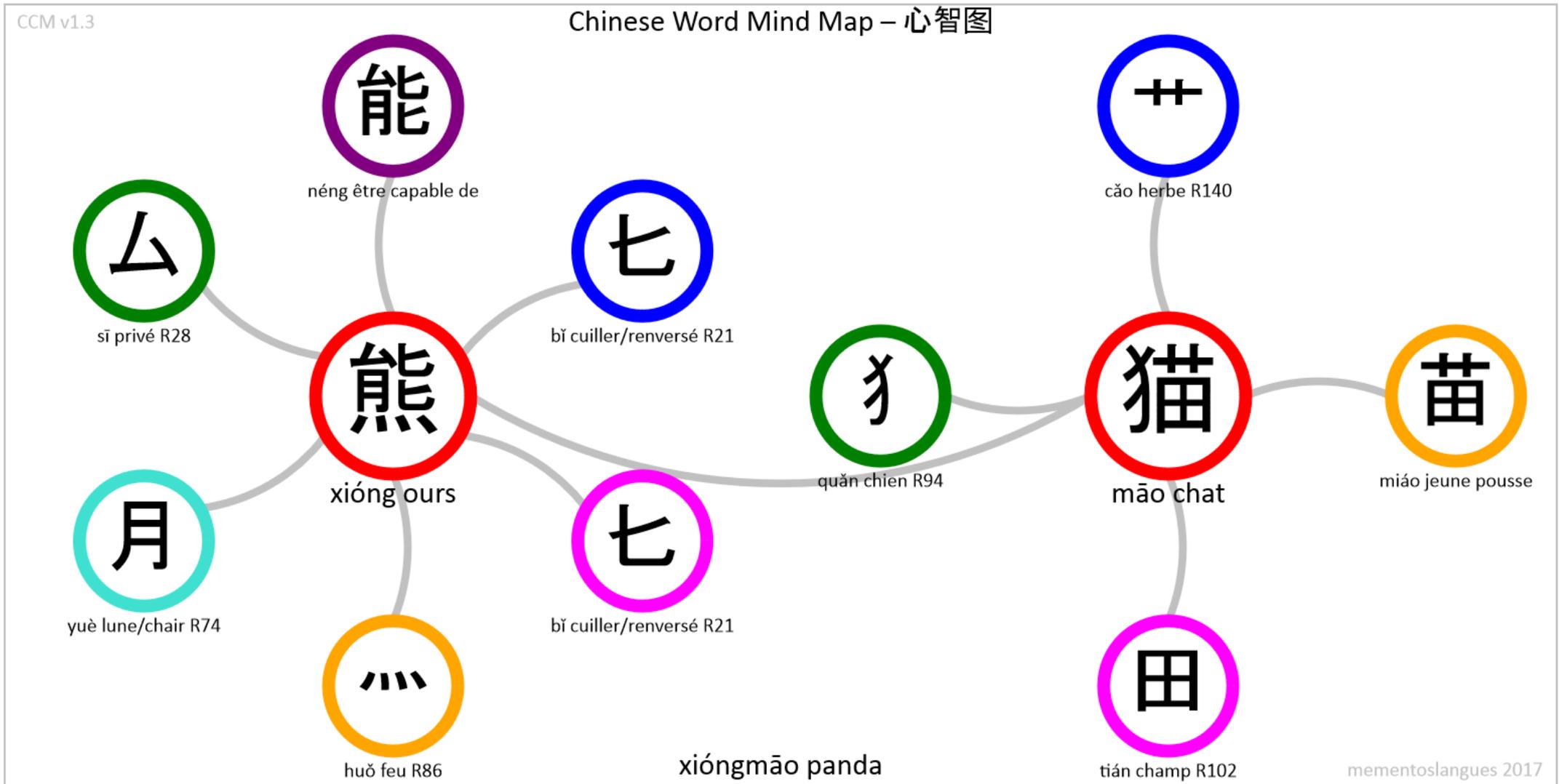


# Chinese Character Mapper

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharactermapper.html>



**Ours:** l'ours ne va pas se priver au clair de lune de renverser deux hommes en sautant par dessus le feu (il en est capable)

**Chat:** poursuivi par le chien, le chat se faufile dans l'herbe en prenant la clé des champs

## Mind Map

Le concept de [Mind Map](#) (ou Carte Mentale) a été popularisé par [Tony Buzan](#). C'est un schéma supposé représenter le fonctionnement de la pensée. Pour être efficace, une carte mentale doit présenter les caractéristiques suivantes:

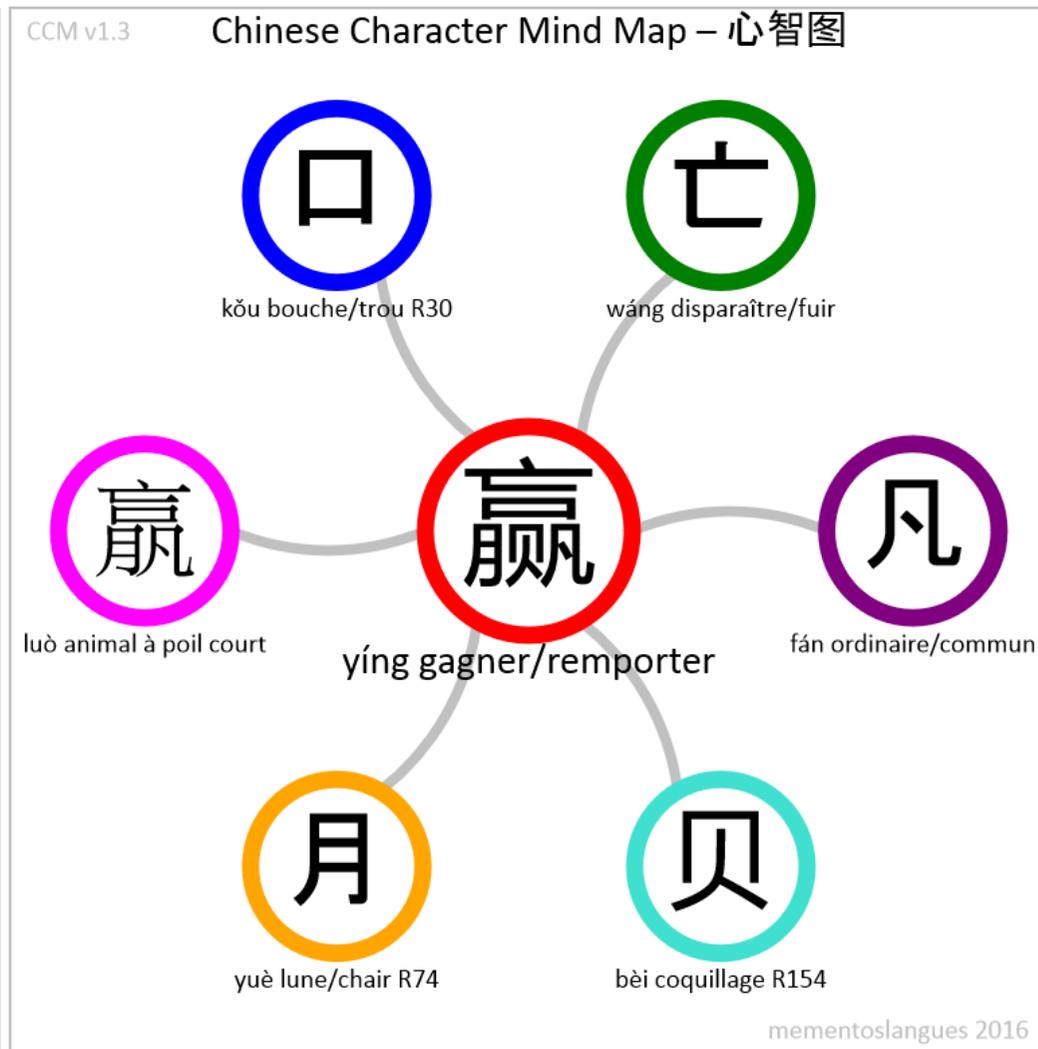
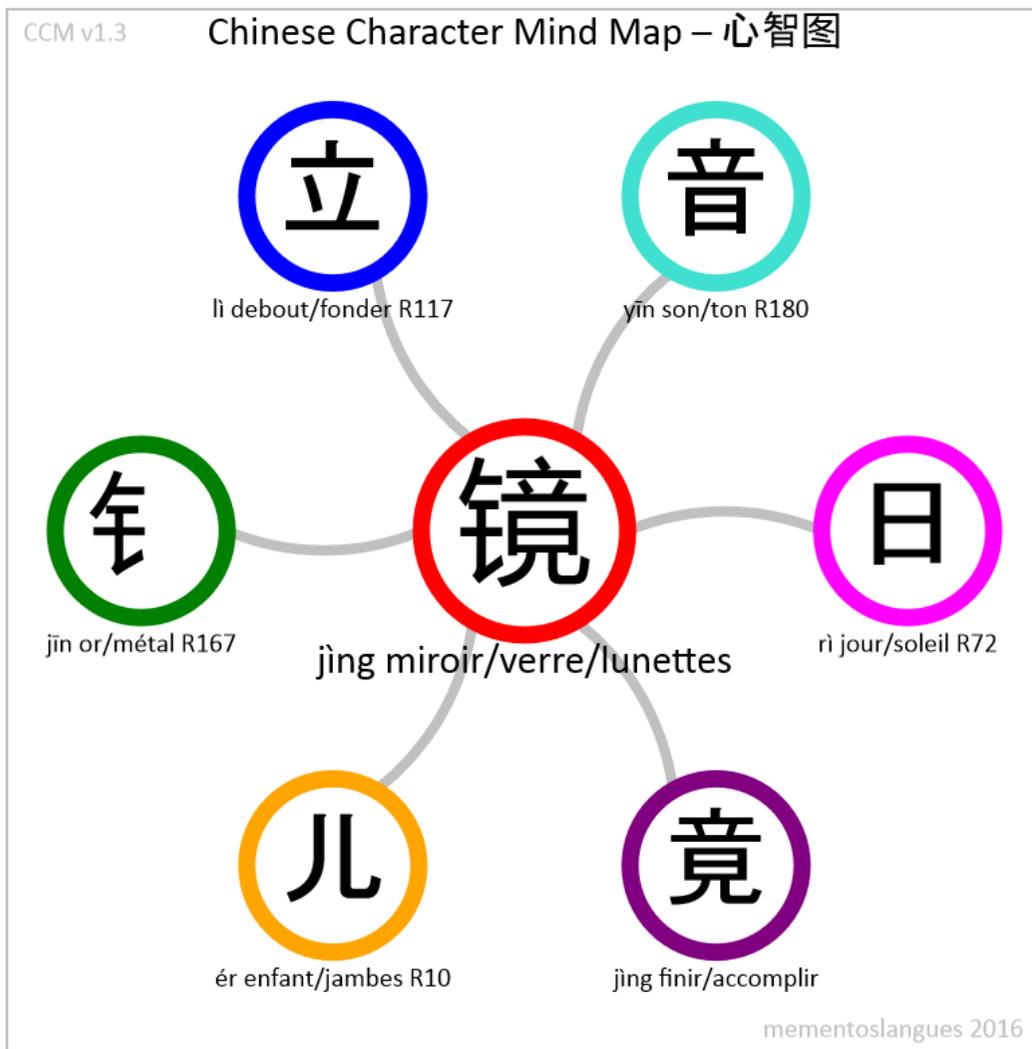
- les informations principales sont entourées de formes arrondies en couleur, *jamais* rectangulaires ni de couleur noire
- ces formes sont connectées par des lignes courbes, *jamais* par des lignes droites
- les données sont échangées entre les formes, en suivant les lignes courbes de connexion

Ce concept peut aisément s'appliquer à la (dé)composition de caractères chinois, ainsi qu'à la formation de mots. Afin de pouvoir retenir et interpréter correctement la signification d'un caractère, il convient de limiter un schéma à 6 composants satellites gravitant autour d'un caractère central, car, depuis les travaux de [George Miller](#), il est maintenant bien établi que la mémoire à court terme ne peut *raisonnablement* retenir que  $6 + 1 = 7$  éléments d'information à la fois.

Ici, le choix du schéma des points de départ des connexions s'est porté sur une horloge horaire affichant les 12 heures, entraînant ainsi une précision de positionnement angulaire de  $30^\circ$ , largement suffisante:



## Exemple 1: caractères complexes



"镜": ["钅", "立", "日", "儿", "音", "竟", 9, 11, 3, 7, 1, 5],

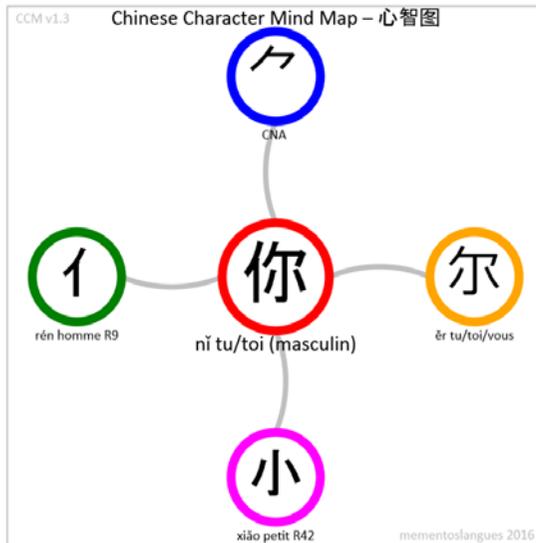
"赢": ["亡", "口", "羸", "月", "贝", "凡", 1, 11, 9, 7, 5, 3],

Une *seule* ligne de code suffit pour décrire entièrement un caractère, même complexe, à condition que les composants existent déjà dans la base de données.

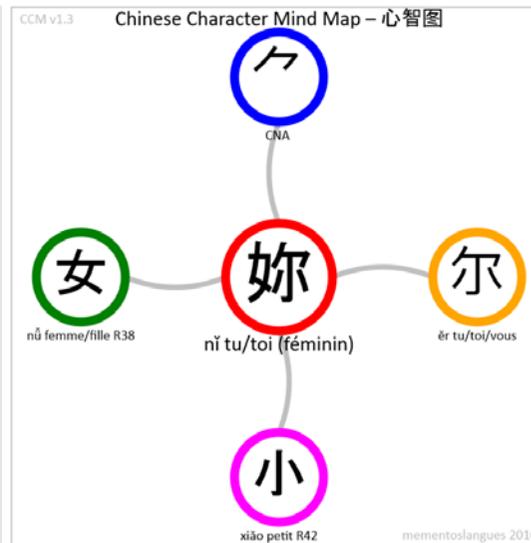
## Exemple 2: pronoms personnels



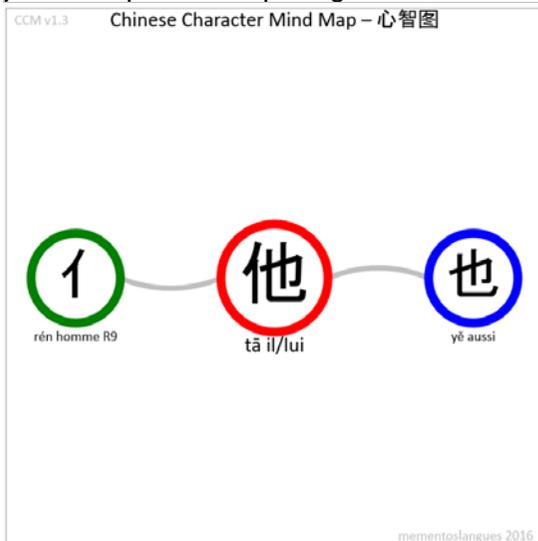
Moi, de ma main armée,  
je suis là pour vous protéger



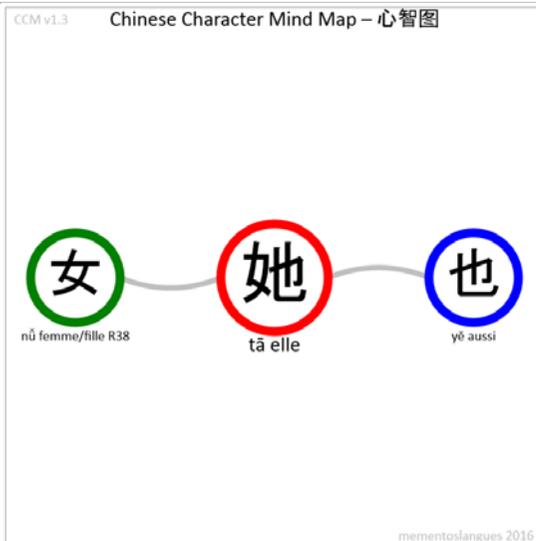
Toi, homme petit sous un toit



Toi, femme petite sous un toit



Lui aussi



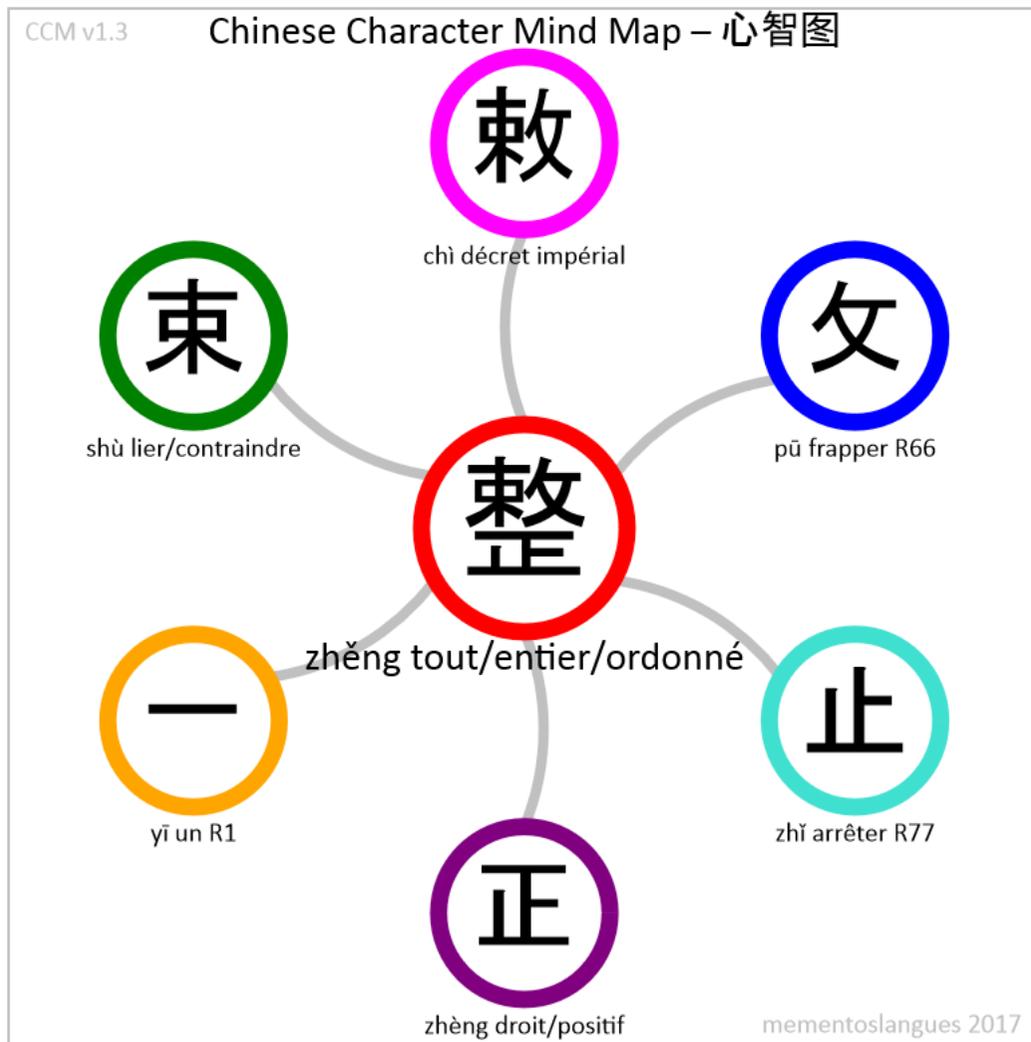
Elle aussi



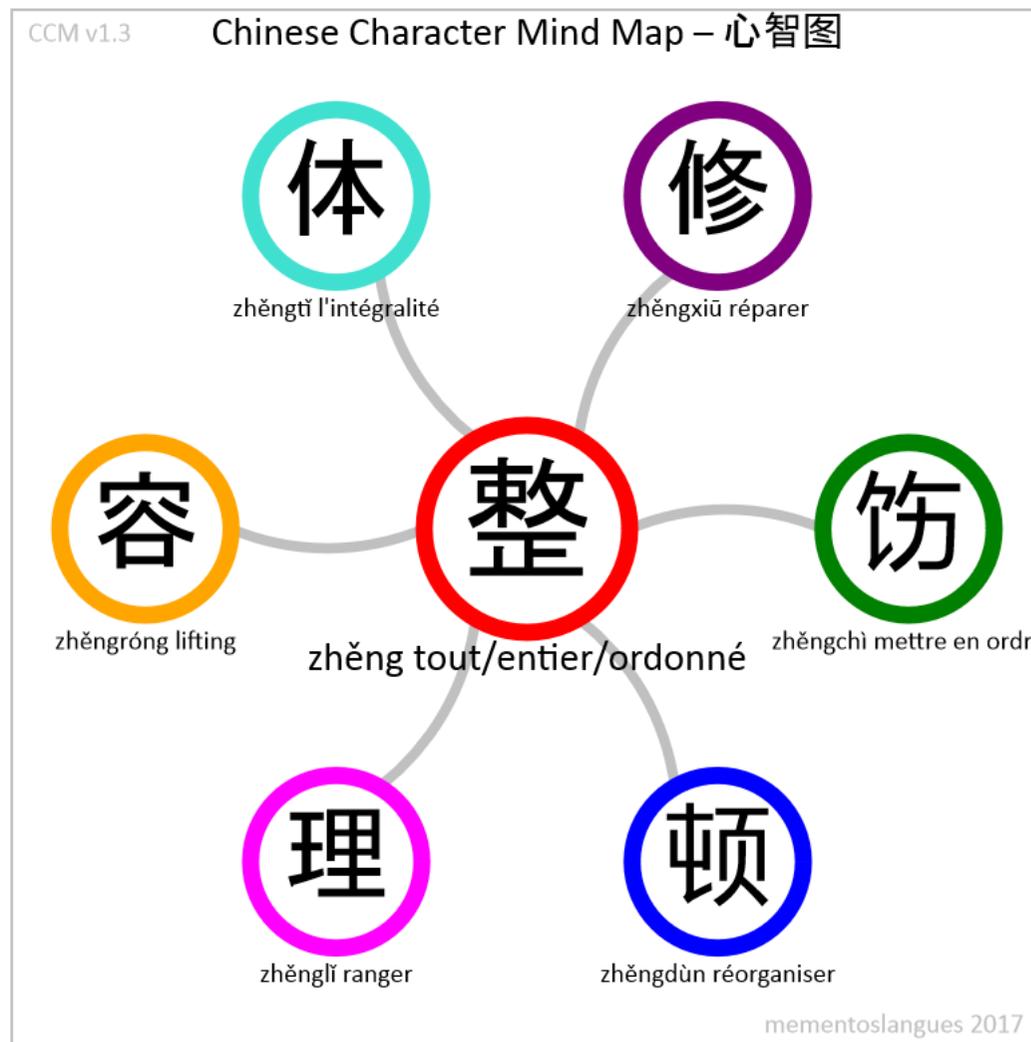
[Félicie aussi](#) – Fernandel, lui il s'en rappelle

Ceci est une application pratique du petit guide 'Tout sur la mémoire' de Tony Buzan: de la rime, du rythme et du lien...et pas plus de 6 mots par ligne !

### Exemple 3: composants, caractères et mots (1)



Composition de composants en *caractères*



Composition de caractères en *mots*

La technique de composition de composants pour obtenir des caractères peut aussi être appliquée très facilement à la composition de caractères pour former des mots, en conservant le principe de 6 éléments satellites au maximum, afin de bien faciliter la mémorisation des éléments. De plus, le développement du logiciel du compositeur de mots est simplifié car il est possible de récupérer une bonne partie du code déjà existant pour le compositeur de composants. Enfin, la présentation graphique globale étant la même dans les deux cas, on bénéficie de la rapidité de la mémoire procédurale déjà mise en place dans le cerveau pour le compositeur de composants. Ci-dessus, c'est bien le *même* caractère qui est à la fois décomposé en composants et associé à d'autres caractères pour former des mots.

## Exemple 4: composants, caractères et mots (2)



### Composition de composants en caractères

### Composition de caractères en mots

Ci-dessus, ce sont deux caractères *différents* qui sont présentés. Le premier est décomposé en composants et le second est associé à d'autres caractères pour former des mots. Ici, les deux caractères principaux ont eux-mêmes été associés pour former un mot supplémentaire. On peut donc constater que le processus de composition de composants est strictement identique à celui de composition de caractères. On pourrait donc suggérer que ce serait aussi le même processus de mémorisation qui serait appliqué par le cerveau dans les deux cas, d'où l'avantage d'utiliser une procédure *unifiée* d'apprentissage à base de *Mind Maps*.

## Base de données

La base de données est contenue dans le fichier JavaScript `chinesecharactermapper_database.js` et comprend 5 tableaux:

- la totalité des composants, incluant les 214 radicaux du dictionnaire *KāngXī*, repérés par la lettre R suivi de leur index: **CCM.componentArray**
- les composants spécifiques au caractère sélectionné ainsi que leur position horaire: **CCM.characterComponentArray**
- la prononciation *pīnyīn* et la traduction en français par caractère: **CCM.characterArray**
- la prononciation *pīnyīn* et la traduction en français par mot: **CCM.wordArray**
- la prononciation *pīnyīn* et la traduction en français par mot-domino: **CCM.dominoWordArray**

Les tableaux peuvent être mis à jour ou augmentés relativement aisément par un utilisateur à l'aide d'un simple éditeur de texte comme le Bloc-notes de Microsoft, ou équivalent. Il faut noter que toutes les listes sont ordonnées suivant la suite des lettres de l'alphabet *pīnyīn*. Quelques exemples:

### CCM.componentArray

```
"匕": ["bǐ", "cuiller/renversé", "R21"],  
"冱": ["jiàng", "tomber/descendre"],  
"マ": ["CNA", ""]
```

### CCM.characterComponentArray

```
"海": ["氵", "宀", "每", "母", 9, 12, 3, 6],  
"熊": ["亠", "匕", "匕", "灬", "月", "能", 10, 12, 4, 6, 8, 2],  
"猫": ["犹", "艹", "田", "苗", 9, 12, 6, 3],
```

### CCM.characterArray

```
"海": ["hǎi", "mer/océan/lac"],  
"猫": ["māo", "chat"],  
"熊": ["xióng", "ours"],
```

### CCM.wordArray

```
"海熊": ["hǎixióng", "otarie"],  
"熊猫": ["xióngmāo", "panda"],
```

### CCM.dominoWordArray

```
"办法": ["bànfǎ", "moyen/méthode/solution/mesure"],  
"法语": ["fǎyǔ", "langue française"],
```



La base de données est accédée en exécutant le fichier HTML `chinesecharactermapper.html` et ce fichier doit comprendre la liste des caractères et mots disponibles, ordonnés suivant la suite des lettres de l'alphabet *pīnyīn*, par exemple:

```
<option value="海">海 hǎi mer/océan/lac
<option value="猫">猫 māo chat
<option value="熊">熊 xióng ours
<option value="熊猫">熊猫 xióngmāo panda
<option value="办法">办法 bànfǎ moyen/méthode/solution/mesure
```

## Dictionnaires

La base de données a été constituée à partir de dictionnaires papier ou en ligne, parmi lesquels on peut noter:

- [Chine nouvelle](#)
- [Wikitionary](#)
- [Written chinese](#)
- [mdbg](#)
- [Pleco](#) (incluant le [Grand Ricci Numérique](#))
- Dictionnaire Chinois-Français de 1990 by The Commercial Press Ltd, toujours aussi utile...

Il faut noter que la base de données *n'est pas* elle-même un dictionnaire car il a fallu limiter le nombre de traductions possibles par caractère. Il convient donc d'utiliser des dictionnaires et des manuels de chinois en complément du logiciel.



## CCM (Chinese Character Mapper)

Le logiciel CCM utilise la *mémoire cache* du navigateur dans lequel il est lancé, Firefox par exemple, pour y installer *temporairement* les fichiers JavaScript dont il a besoin pour fonctionner. La présence du fichier **chinesecharactermapper.APPCACHE** permet d'assurer la mise à jour *automatique* du logiciel par comparaison des contenus des deux fichiers APPCACHE, celui situé sur le serveur distant et celui chargé sur la machine locale de l'utilisateur, lorsque celui-ci est connecté à Internet. La présence du fichier APPCACHE permet en outre de continuer à utiliser le logiciel, même quand l'utilisateur n'est *plus* connecté à Internet, à condition de ne pas avoir effacé la mémoire cache du navigateur.

Le logiciel fonctionne absolument sur *toutes* les machines disposant d'un navigateur Internet, pourvu que celui-ci ait été autorisé à exécuter du code JavaScript, ce qui est normalement autorisé par défaut. Il peut donc être utilisé en particulier sur tablettes et téléphones portables, connectés ou non à Internet. JavaScript étant un langage interprété, aucun compilateur de code source n'est alors nécessaire.

Le développement logiciel en JavaScript est facilité par l'utilisation de produits gratuits comme [Visual Studio Express](#) de Microsoft pour l'édition du code et [JSLint](#) pour le débogage.

La version actuelle CCM v1.3 contient l'ensemble des caractères et des mots (à 2 caractères) du vocabulaire des [HSK 1](#), [HSK 2](#), [HSK 3](#) ainsi que ceux du Tome 1 de la méthode [Le chinois...comme en Chine](#) de [Bernard Allanic](#). Des [mots-dominos](#) chinois ont également été rajoutés.



## Chinese Character Viewer

Si l'on souhaite afficher *directement* quelques caractères ou mots, il n'est pas alors nécessaire d'exécuter le **Mapper**, mais plutôt le **Viewer**. Il suffit alors de rajouter le point d'interrogation (?), suivi du caractère ou du mot à décomposer à la suite du nom du fichier HTML *complet*, puis de cliquer sur le lien ainsi créé. Il existe deux types de liens possibles, soit un lien hypertexte long *inséré* par un outil de traitement de texte dans un *alias* court, soit un lien complet. Voici quelques exemples:

### Liens masqués:

Pour s'adresser à une personne de sexe féminin:

[妳好](#)

[妳 怎么样 ?](#)

[再见](#)

Pour s'adresser à une personne de sexe masculin:

[你好](#)

[你 怎么样 ?](#)

[再见](#)

Pour vérifier la base de données, cliquer ici: [check](#)

### Liens complets:

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?妳好>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?妳>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?怎么>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?样>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?再见>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?你好>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?你>

<http://www.mementoslangues.fr/Chinois/ChineseCharacterMapper/chinesecharacterviewer.html?check>

