AMERINDIA n°17, 1992

# nahuatl, informatique et *TEMOA*

Marc THOUVENOT

*CNRS-URA 1026*

La transcription du nahuatl à l'aide des caractères latins, depuis le début du XVIe siècle, alors que son écriture traditionnelle était pictographique, peut sembler à priori un facteur tout à fait favorable aux traitements informatiques de cette langue.

C'est effectivement le cas, mais il demeure cependant une certaine incompatibilité entre le nahuatl et les ordinateurs. Ces machines ont en effet une prédilection pour la régularité, alors que la caractéristique formelle la plus évidente du nahuatl écrit c'est l'absence d'une norme unique pour sa transcription[1].

En français contemporain pour un mot donné, à quelques exceptions près, il n'existe qu'une orthographe. C'est cette unicité qui permet de le trouver aisément en utilisant un dictionnaire traditionnel ou bien un dictionnaire stocké

[1] La langue nahuatl n'y est pour rien. Dès le premier contact entre les Européens et les Aztèques leur langue, le nahuatl, a été transcrite en caractères latins et dès cet instant un même mot a été rendu par diverses graphies. La convergence d'au moins quatre facteurs explique ce phénomène : l'écriture en caractères latins n'est que partiellement phonétique ; certains sons du nahuatl étaient inconnus des Espagnols ; à cette époque on admettait beaucoup plus facilement qu'aujourd'hui qu'un même mot puisse connaître plusieurs orthographes ; enfin le degré d'éducation européenne et la sensibilité de l'oreille de celui qui tenait la plume n'étaient sans doute pas sans influence.

sur un support informatique ou encore la fonction "Cherche" de son traitement de texte préféré. Ainsi le mot "maison" n'est-il jamais écrit *méson, *mêson, *meison, *mèson, *mayson ou encore *maisong, *mésong, *mêsong, *meisong, *mèsong, *maysong ou encore *mèzon, *mayzon, *maizong, *mézong, *mêzong, *meizong, *mèzong, *mayzong.

En nahuatl la multiplicité est la règle et l'unicité l'exception. Ainsi un mot simple comme **ihuan** : "et" peut apparaître sous les dix-neuf formes attestées suivantes : **ihua / ihuan / ioâ / ioan / iuan / ivâ / ivan / jhoan / joan / juan / jvan / yhua / yhuâ / yhuan / yoa / yoâ / yoan / yuan / yvan**[2].

Le nombre important de graphies s'explique évidemment parce que **ihuan** a été relevé dans quatorze documents différents. Mais il ne faudrait cependant pas croire que l'unicité règne à l'intérieur de chacun des textes. Loin s'en faut. Ainsi les Annales de Tlatelolco offrent huit orthographes différentes.

Pourquoi cela pose-t-il un problème aux machines alors qu'un lecteur, ayant un tant soit peu l'expérience des textes nahuatl, rencontrant l'une quelconque de ces formes sait qu'il s'agit là du mot **ihuan** : "et" ? Il me semble que cela tient au fait que l'homme travaille avec en tête la réalisation phonique d'un mot alors que la machine utilise la seule forme qu'elle connaisse aujourd'hui c'est-à-dire sa forme graphique. Si je cherche le mot **ihuan** dans un texte, je cherche en réalité /***w²n**/ quelle que puisse être sa forme. En entrée, pour prendre la terminologie informatique, j'ai du son et en sortie j'ai toujours du son

---

[2] Ceci sans tenir compte du fait que la lettre initiale peut être une lettre majuscule ou minuscule.
*Annales de Tlatelolco* : **yhua / yhuâ / yhuan / yoâ / yoa / yoan / ihuan / ihua** ; Corpus *Techialoyan* : Textes en caractères latins : **ihuan / iuan** ; *Troisième Relation de Chimalpahin* : **ihuan / yhuan / yhuâ / yuan / iuan** ; *Annotations du Codex Xolotl* : **ihuan / ivâ** ; *Crónica Mexicayotl* : **yhuan / ihuan** ; Ecrits de Cristobal del Castillo : **ihuan** ; BN 303 ou *Anales Mexicanos* : **ihuan** ; *Histoire mexicaine depuis 1221 jusqu'en 1594* : **yhuan / yhuâ** ; *Fragment de l'histoire des anciens mexicains* : **yhua / yhuan** ; *Fragment d' histoire du Mexique* : **yhuâ / yoâ / yhuan** ; *Annales de Cuauhtitlan* : **ihuan / yoan / yoa / yhuan / yhua / ihua** ; *Leyenda de los Soles* : **ihuan** ; *Annotations de la Mapa Tlotzin* : **ihuan** ; *Codex de Florence* : **ioâ / ioan / iuan / ivan / jhoan / joan / juan / jvan / yoâ / yoan / yvan**

Toutes ces listes ont été obtenues à partir des paléographies publiées par les Editions SUP-INFOR, à l'exception de la dernière concernant le *Codex de Florence* qui est extraite d'un article de R. Joe Campbell et Mary L. Clayton : "Bernardino de Sahagun's Contributions to the Lexicon of Classical Nahuatl", dans *The Work of Bernardino de Sahagun*. Edité par J. Klor de Alva, H. B. Nicholson and Eloise Quiñones Keber, p. 304. Institute for Mesoamerican Studies, the University at Albany. (Le mot **yvan**, qui figure dans le livre XI, a été rajouté. Ceci grâce au fait que Marc Eisinger (IBM France) m'a prêté la version informatique de ce livre).

mais matérialisé par une graphie variable. Cette variabilité tenant simplement au fait que les caractères latins sont un système de notation seulement partiellement phonétique.

Quand on est aidé d'une machine pour chercher des mots il en va tout autrement. La machine n'acceptant en entrée que des mots écrits et non pas des sons il faut lui fournir une graphie qu'elle va ensuite tenter de retrouver dans les textes. Pour l'ordinateur un mot n'est qu'une suite de lettres. Une recherche est un succès si à la première lettre du mot cherché correspond la première lettre d'un mot du texte et si à la deuxième lettre du mot cherché correspond la deuxième lettre et ainsi de suite jusqu'à la dernière lettre du mot donné en entrée. Si l'on recherche **ihuan**, l'ordinateur ne trouve que les mots composés de **i + h + u + a + n**. Toute lettre étrangère à cette série provoque un échec.

Pour surmonter cette façon bornée de travailler qu'ont les ordinateurs il existe une solution simple : fournir à la machine pour chaque mot toutes ses graphies possibles. Mais cette fois on se heurte aux limites humaines. S'il n'est pas très compliqué de percevoir la similarité de formes différentes, par contre l'opération inverse qui consiste, à partir d'une forme, à imaginer toutes les autres est beaucoup plus complexe et sans doute réservée à quelques spécialistes chevronnés !

La multiplicité des orthographes n'est pas le seul problème que l'on rencontre sur la voie de l'exploitation de textes nahuatl par un ordinateur. Une autre difficulté majeure tient au fait que la segmentation des mots n'est pas constante. Ainsi, par exemple, dans un des documents conservé par la Bibliothèque Nationale de Paris trouve-t-on dans la même phrase un personnage dont le nom est une fois écrit **Cuauh xilotl** et une autre **Cuauhxilotl**. Parfois même des signes de ponctuation peuvent se glisser dans ces blancs. Ainsi quelques lignes plus loin peut-on lire le mot **cuahuitl** écrit **Cua, huitl**[3].

Une dernière difficulté tient au fait que les paléographies de textes anciens peuvent parfois introduire à l'intérieur même d'un mot des signes qui perturbent les recherches. Ainsi lorsque qu'un mot est en partie illisible et est reconstitué, la partie hypothétique est mise entre crochets. Ou bien quand un mot est coupé par un glyphe cette coupure est matérialisée par deux diagonales.

---

[3] *P094A* : Tola, Santa Isabel, Títulos de tierras pertenecientes al pueblo de. folio 9 verso.

Ainsi **Cuauhxilotl** peut parfaitement avoir la forme **Cuauh//xil[o]tl** dans une paléographie.

Trois facteurs, l'orthographe variable, la segmentation aléatoire et les signes paléographiques, concourent donc à créer la pluralité là où l'on souhaiterait en fait la simplicité d'une forme unique !

L'unique graphie à laquelle on pense généralement au moment de faire une recherche a des chances de se trouver sous un nombre considérable de formes dans les textes. Ainsi l'anthroponyme **Cuauhxilotl** cité précédemment peut s'écrire théoriquement d'au moins 8192 façons, en ne tenant compte que du premier facteur, les variations orthographiques. Ceci parce que **Cuauh** peut s'écrire aussi bien **Quauh** ou **Quau** ou **Cuau** ou **Quav** ou **Cuav** ou **Quavh** ou **Cuavh** ou **Cuahu** ou encore **Quahu** tandis que **xilotl** peut être orthographié **xilutl** ou **xillutl** ou encore **xillotl**.

D'un côté on a donc une forme en tête, que l'on communique à l'ordinateur, de l'autre la machine doit comprendre que 8192 graphies différentes dans les textes peuvent correspondre au mot recherché.

Comment surmonter cette difficulté ? Il existe principalement deux réponses l'une, traditionnelle, qui consiste à normaliser l'orthographe et la segmentation des paléographies, l'autre qui prend pour principe que la forme originelle des textes doit être respectée et que c'est à la machine de s'adapter aux difficultés. La première attitude, même si elle n'est pas générale, est traditionnelle dans la mesure où on peut l'observer dans un grand nombre de paléographies publiées entre la fin du siècle dernier et aujourd'hui même. La seconde, adoptée pour toutes mes paléographies, est animée par un principe simple. Il faut conserver, dans toute la mesure du possible, les caractéristiques du document original. Aussi bien pour ce qui concerne l'orthographe, que la segmentation, que la ponctuation et la disposition en paragraphes. Dans le cas du nahuatl ce principe peut être d'autant plus facilement observé qu'il y a fort peu de signes qui ne sont pas directement reproductibles par un ordinateur. Ce principe posé il faut trouver un moyen de faire comprendre à une machine qu'un même mot (chaîne de caractère particulière comprise entre un blanc et un autre blanc ou un signe de ponctuation) peut avoir de multiples réalisations graphiques.

C'est à cette fin qu'ont été conçus *TEMOA*[4], logiciel de recherche sur les chaînes de caractères dans les textes nahuatl, et *GENOR*, générateur d'orthographe auquel est dévolue la charge de résoudre le problème des orthographes multiples.

## *GENOR*

La philosophie générale de *GENOR* tient en trois points :

- pour un mot donné *GENOR* doit pouvoir générer toutes les formes nécessaires pour le retrouver dans tous les textes nahuatl paléographiés selon les principes évoqués ci-dessus.

- on doit obtenir la même liste de mots générés quelque soit celui donné en entrée. Ceci afin que chacun puisse conserver ses habitudes orthographique. En pratique si on a l'habitude de travailler surtout sur le *Codex de Florence* pour trouver **ihuan**, on donnera en entrée plutôt **yoan**, alors que si on est un fidèle lecteur de Chimalpahin on demandera plus vraisemblablement le mot **yhuan**. Dans les deux cas on obtiendra la même liste de graphies.

- on accepte que *GENOR* génère des graphies théoriquement possibles mais qui n'existent pas, à condition que le nombre de ces formes ne soit pas trop important.

Ce dernier point peut paraître anodin, alors qu'en fait il est essentiel. Il est clair que peu importe que quelques graphies inexistantes soient générées. C'est sans importance car cela ne modifie pas les résultats obtenus et affecte seulement un peu les temps de réponse. Le problème vient du fait que le principe même de la génération tend inexorablement vers les grands nombres. Il faut savoir que chaque fois qu'une lettre peut se substituer à une autre on multiplie par deux le nombre des formes. Il s'agit d'une progression géométrique dont la

[4] *TEMOA* est un éditeur de texte trilingue (français, espagnol et anglais) doté de fonctions évoluées de recherches de chaînes de caractères, dont certaines sont spécifiques à la langue nahuatl. *TEMOA* permet de faire des recherches sur une, deux ou trois chaînes dans le contexte du mot, de la phrase ou du paragraphe, ou hors contexte sur un nombre infini de chaînes. Il permet l'utilisation de Thésaurus (liste de mots à partir desquels les recherches sont faites) et de Corpus (liste de Documents sur lesquels les recherches sont effectuées). Edité par SUP-INFOR, Paris.Version 2.1, 1992.. (ISBN 2-908782-08-1)

principale caractéristique est d'être incroyablement rapide[5]. Dix substitutions engendrent plus de mille formes et vingt plus d'un million !

Les lignes qui suivent expliquent comment, par l'analyse des caractéristiques de l'écriture du nahuatl, l'application du principe simple de la génération est possible.

Il a été dit précédemment que le mot **cuauhxilotl** pouvait, du fait de la combinaison des substitutions possibles, avoir plusieurs milliers de formes. Pour retrouver toutes ces formes *GENOR* n'a besoin de n'en générer que huit :

**CUAUXILLOTL CUAUXILLUTL CUAUXILOTL CUAUXILUTL QUAUXILLOTL QUAUXILLUTL QUAUXILOTL QUAUXILUTl**[6].

Ceci n'est possible que parce que, à différents stades, il y a application de règles qui peuvent être regroupées en quatre catégories : expansion, contraction, transformation et génération.

## 1) Règles d'expansion :

Le principe général guidant les paléographies des textes est le respect le plus total de la forme originelle. Il est cependant un point sur lequel j'apporte une modification superficielle c'est le développement des abréviations.

Bien souvent on trouve dans les textes nahuatl anciens des abréviations du type **ôcâ** pour **oncan**. Ce type de notation, qui n'est pas liée à la langue nahuatl mais correspond à une pratique courante des écrivains européens[7] de cette époque, concerne non seulement les lettres **â** et **ô** mais encore **ê**, **î**, **û**, **y**, **g.**, **q'** et **µ.**

Les lettres **â**, **ê**, **î**, ou **ô** -abréviations de **an**, **en**, **in** ou **on**- on été transcrites par **ân**, **ên**, **în** et **ôn**. La lettre **y** surmontée d'un accent circonflexe a

---

[5] L'application de toute nouvelle règle multiplie par deux le nombre de formes existant précédemment. La progression, géométrique, est la suivante : 1 Règle -> 2 Formes ; 2 R -> 4 F ; 3 R -> 8 F ; 4 R -> 16 F ; 5 R -> 32 F ; 6 R -> 64 F ; 7 R -> 128 F ; 8 R -> 256 F ; 9 R -> 512 F ; 10 R -> 1 024 F ; 11 R -> 2 048 F ; 12 R -> 4 096 F ; 13 R -> 8 192 F ; 14 R -> 16 384 F ; 15 R -> 32 768 F ; 16 R -> 65 536 F ; 17 R -> 131 072 F ; 18 R -> 262 144 F ; 19 R -> 524 248 F ; 20 R -> 1 048 576 F.

[6] Pour obtenir ces huit formes le niveau 2 de *GENOR* est utilisé.

[7] Dans le dictionnaire de Molina on trouve par exemple "hincapié" écrit "hîcapie" (pour être exact il s'agit en fait d'un tilde sur le i) f. 69 recto. Molina, Fray Alonso de. 1970. *Vocabulario en lengua castellana y mexicana, y mexicana y castellana*, Porrúa, México.

du être remplacée par un **ÿ** et cette lettre a été transcrite par la formule développée **ÿn**. La notation **g.** (sorte de **q** majuscule ressemblant à un **g**) - abréviation de **que-** est transcrite par **gue**. Ainsi le mot **huehuetgue** correspond dans l'original à **huehuetg**. Les **q'** sont transformés en **quï**, tandis que les **µ** le sont en **quë**.

Ces expansions modifient très légèrement l'apparence du texte, mais d'une part elles facilitent grandement les recherches et d'autre part la présence des accents permet de rétablir, si besoin est, la forme originelle. Ces modifications sont dites superficielles dans la mesure où la simple observation du texte permet, si besoin est, de reconstituer la forme initiale.

En application de ces règles le mot **ioâ** est automatiquement modifié en **ioân** dans le texte même.

## 2) Règles de contraction :

Il existe trois règles de contraction, qui toutes trois sont appliquées, aussi bien aux chaînes recherchées qu'aux mots des textes, en mémoire.

1) Toute chaîne de caractères comportant des signes paléographiques est réduite par suppression de ces signes. En vertu de ce principe **Cuauh//[x]ilotl** est réduit en **CUAUHXILOTL**.

2) Toute chaîne de caractères comportant en son sein des caractères séparateurs (blanc ou signes de ponctuation) est réduite à sa forme sans séparateurs. Ainsi **Cuauh, xilotl** est réduit en **CUAUHXILOTL**.

3) Toute chaîne de caractères comportant en son sein la lettre **h**, sauf si elle est située immédiatement après la lettre **c** formant ainsi le digraphe **ch**, est réduite par suppression des **h**. Ainsi **Cuauhxilotl** est réduit en **CUAUXILOTL**. La lettre **h** est simplement considérée comme une lettre "muette".

L'application successive de ces trois règles de contraction réduirait, en mémoire, un mot écrit dans un texte **Cuauh, //[x]ilotl** en **CUAUXILOTL**. Ceci explique que le nombre de formes à générer soit tout à fait inférieur aux formes figurant réellement dans les paléographies. Il existe une autre raison qui tient à la mise en oeuvre des règles de transformation.

## 3) Règles de transformation :

On voit avec l'exemple de **ihuan** que certaines lettres se substituent aux autres. Ainsi les lettre **j** et **y** viennent parfois remplacer le **i**, de la même façon à la place du **u** on peut trouver un **v** : **ihuan = juan = yhuan** et **yuan = yvan**.

Dans tous les cas où il existe une relation univoque entre des lettres celles-ci sont automatiquement substituées en mémoire. C'est ainsi que les **v** ou **V** sont transformés en **U**. Les **ç** en **Z**. Les **y**, **ÿ** ou **Y** en **I**. Les **j** ou **J** en **I**. Les **î**, **ï**, **í** et **ì** en **I**. Les **â** et **ä** en **A**. Les **ô** et **ö** en **O**. Les **ê** et **ë** en **E**. Les **g** ou **G** en **Q**. Les **s** ou **S** en **X**. Ces règles découlent directement du travail de Marc Eisinger sur les orthographes du *Codex de Florence*[8].

Ces transformations portent toujours sur une seule lettre et seulement si une relation d'égalité l'unit toujours à la lettre de substitution. Ainsi **i** est substitué à **j** parce que **j = i** est toujours vrai. A l'inverse **o = u** (ici **yoan = yuan**) n'étant vrai qu'occasionnellement il n'existe pas de règle de transformation pour les **o** et les **u**.

Dans tous les cas où les substitutions de lettres sont conditionnelles ou bien concernent plusieurs lettres, ce sont alors les règles de génération qui sont chargées de cette tâche, par l'intermédiaire de *GENOR*.

## 4) Règles de génération :

*GENOR* est un système expert qui appliquant des règles à une chaîne de caractères introduite par un utilisateur génère toutes les graphies théoriquement possibles. Comme tout programme de ce type il comporte d'un côté un moteur, appelé moteur d'inférence, et de l'autre une base de règles. Il ne sera pas question ici du moteur mais seulement des règles[9].

Les règles présentent quelques traits généraux. Elles sont réversibles, générales, graduées et généralement conditionnelles.

---

[8] EISINGER, Marc. 1977. *Codex de Florence et informatique. Propositions pour l'étude systématique des textes nahua*. Paris : Mémoire de l'Ecole des Hautes Etudes en Sciences Sociales, 87 pp. La liste des transformations se trouve en page 20. Depuis, au fil des années, elle a été enrichie. Voir *Indexing the Florentine Codex* (manuscrit).

[9] *GENOR* est un système expert dont le moteur d'inférence travaille en chaînage avant, en logique des propositions pures, de façon monotone. Les bases de faits et de règles sont écrites en mode texte tandis que le moteur est un exécutable écrit en langage C.

Elles sont réversibles car chaque conversion d'une lettre ou syllabe est accompagnée de son contraire. Ainsi la règle och -> uch est indissociable de la règle contraire uch -> och. Ceci est indispensable pour obtenir les mêmes formes générées quelque soit le mot entré. Ainsi **tenochca** donne **tenuchca** et inversement **tenuchca tenochca**.

Les règles sont d'une application générale dans la mesure où toute lettre ou syllabe répétée à l'intérieur d'un mot sera modifiée. C'est en vertu de ce principe et de la règle l <-> ll (si compris entre deux voyelles) que le mot **cholula** prendra les formes **chollula**, **cholulla** et **chollulla** (sans parler des autres formes générées en application d'autres règles).

Les règles sont graduées dans la mesure où elles ne sont pas toutes forcément appliquées. Il existe trois niveaux, correspondant à trois grands styles de variations orthographiques. Ces trois grandes familles sont : le style Classique, le style Codex de Florence et enfin le style Techialoyan. Ces trois niveaux sont hiérarchisés. Ainsi les règles de génération du style Techialoyan comprennent celles propres à ce style, plus celle du style Codex de Florence, qui elles mêmes incluent celles du style Classique.

Les règles sont généralement conditionnelles car à côté d'un certain nombre d'entre elles qui s'appliquent toujours, il en est d'autres qui ne sont mises en oeuvre que dans le cadre de certaines limites contextuelles.

A la différence des règles de contraction et de transformation, qui s'appliquent aussi bien aux chaînes données en entrée qu'à celles figurant dans les paléographies, les règles de génération ne concernent que celles données en entrée.

Ces règles ont été dégagées en utilisant conjointement les sources elles-mêmes et quelques études existantes. Les sources sont l'ensemble des documents

paléographiés et publiés sur support informatique[10], *L'index du Codex de Florence* de M. Eisinger[11] et la liste des annotations relatives au *Codex Xolotl*[12]. Dans le temps ces sources couvrent une période allant de la première moitié du XVI° jusqu'à la seconde moitié du XVIII°, les dates extrêmes étant représentées par les *Annales de Tlatelolco* (1528) et les textes des *Codex Techialoyan*. Les études utilisées sont celles de R. Andrews[13], M. Launey[14] et Th. Sullivan[15].

---

[10] *CASTILLO* : *Ecrits de Cristobal del Castillo*. Marc Thouvenot. 1990. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 263, 305 et 306. (62K)
*303PBN* : *BN 303 ou Anales Mexicanos*. Marc Thouvenot. 1990. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 303. (20K)
*3CHIMAL* : *Troisième Relation de Chimalpahin*. Jacqueline de Durand-Forest avec la collaboration de Marc Thouvenot. 1990. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 74. (154K)
*P001A* : *Annotations du Codex Xolotl*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 1-10. (31K)
*P011A* : *Annotations de la Mapa Quinatzin*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 11-12. ((19K)
*P022B* : *Annales de Tlatelolco*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 22bis. (125K)
*P040A* : *Histoire mexicaine depuis 1221 jusqu'en 1594*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 40. (32K)
*P085A* : *Fragment de l'histoire des anciens mexicains*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 85. (24K)
*P217A* : *Fragment d' histoire du Mexique*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 217. (44K)
*P311A* : *Crónica Mexicayotl*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 311. (160K)
*P312A* : *Codex Chimalpopoca : Annales de Cuauhtitlan*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 312. (215K)
*P312B* : *Codex Chimalpopoca : Leyenda de los Soles*. Marc Thouvenot. 1992. Manuscrit nahuatl : Bibliothèque Nationale de Paris N° 312. (47K)
Tous les documents ci-dessus forment le corpus de textes dits Classiques utilisés dans la détermination des règles de niveau 1.
*TECHIA* : *Corpus Techialoyan : Textes en caractères latins*. Joaquín Galarza avec la collaboration de Marc Thouvenot. 1990. 43 Manuscrits nahuatl. (590K)
Les quarante-trois documents *Techialoyan* ci-dessus forment le corpus des textes dits *Techialoyan* utilisés dans la détermination des règles de niveau 3.

[11] EISINGER, Marc. (en préparation). *L'Index du Codex de Florence*, SUP-INFOR, Paris. Il s'agit de la liste de tous les mots du *Codex de Florence* avec pour chacun d'eux tous leurs emplacements dans le texte. Dans cet index toutes les orthographes originales ont été conservées. Marc Eisinger m'a confié cet index avant sa publication, qu'il en soit remercié. C'est cet index qui a permis la détermination des règles de niveau 2.

[12] THOUVENOT, Marc. 1987. *Codex Xolotl. Etude d'une des composantes de son écriture : les glyphes. Dictionnaire des éléments constitutifs des glyphes.* Thèse pour le doctorat d'Etat ès-Lettres et Sciences Humaines. Ecole des Hautes Etudes en Sciences Sociales. A l'occasion de ce travail j'ai réuni, autour des glyphes du *Codex Xolotl*, toutes les annotations, gloses et citations s'y rapportant. C'est cette liste de plus de 3000 éléments qui m'a fourni le point de départ pour l'établissement des règles de génération de *GENOR*.

[13] ANDREWS, Richard. 1975. *Introduction to Classical Nahuatl*. University of Texas Press. pp. 406-409.

[14] LAUNEY, Michel. 1979. *Introduction à la langue et à la littérature aztèques.* Paris, L'Harmattan. Tome 1, pp. 11-18 et Tome 2, pp. 408-414.

Sachant que les principes exposés par les spécialistes du nahuatl n'étaient pas absolument complets il fallait essayer de trouver un moyen pour découvrir les règles n'ayant pas été mises à jour[16]. La pratique de la paléographie permet de prendre conscience de nouvelles graphies mais c'est surtout un travail systématique, informatisé, sur les textes qui a permis de faire sortir les règles. Pour y parvenir trois index ont été constitués. L'un pour les textes classiques, un autre pour le *Codex de Florence* et un dernier pour les textes *Techialoyan*. Lors de la création de ces index les règles de transformation et de contraction ont été appliquées. Chacune de ces trois listes de mots est organisée en un fichier comportant deux champs. L'un contient le mot et l'autre sa conversion par une fonction du type soundex()[17] qui convertit un mot en un nombre censé représenter sa composition phonétique. Les mots sont alors indexés sur les nombres et c'est la consultation du fichier ainsi ordonné qui permet, parfois, de voir surgir les règles. Si ce moyen a permis d'en trouver un certain nombre, il ne s'agit pas pour autant d'une panacée universelle. Ceci pour au moins trois raisons : la segmentation irrégulière des mots, le caractère agglutinant de la langue et les flexions. Malgré tout, cela a permis d'enrichir la liste des variations orthographiques possibles dans les textes nahuatl anciens.

On trouvera à la suite la synthèse des règles existantes, celles-ci ayant été regroupées par grands principes et ordonnées en voyelles, consonnes et mélange des deux.

---

15 SULLIVAN, Thelma D. 1976. *Compendio de la gramática náhuatl*. México, U.N.A.M. pp. 23-27.

16 Situation étrange dans le contexte d'un système expert où normalement on met à profit le savoir des experts pour créer les règles. Là il convient de créer le savoir pour ensuite le formaliser et alimenter le système !

17 Cette fonction utilise un algorithme créé par Donald E. Knuth dans *The Art of Computer Programming,* Volume 3, "Sorting and Searching", page 392. Cette fonction a pour but de permettre de trouver des mots phonétiquement similaires mais aux graphies différentes.

## Voyelles

| Principes | Niveaux + Règles | Exemples[18] |
|---|---|---|
| a <-> e[19] | 0 | **tlachia** = **tlachie** (*CF*) |
| e <-> i[20] | 1 tel <-> til | **tlatelulca = tlatilulca** (*P312B*) |
| e <-> ie[21] | 2 e <-> ie | **eecatl = yehecatl** (*CF*) |
| i <-> ii[22] | 1 i <-> ii | **quiahuitl = quiyahuitl** (*P312B*) |
| o <-> u[23] | 1 coio <-> cuio | **coyohuacan = cuyohuacan** (*3CHIMAL*, *P311A*) |
| | 1 coco <-> cuco | **tetzcoco = Tetzcuco** (*3CHIMAL*) |
| | 1 eoa <-> eua | **iehoantin = iehuantin** (*P022B*, *3CHIMAL*) |
| | 1 o <-> u | **yoan = yhuan** (*P022B*) |
| | 1 oca <-> uca | **tetzcoca = tetzcuca** (*3CHIMAL*) |
| | 1 och <-> uch | **tlacochcalcatl = tlacuchcalcatl** (*P022B*) |
| | 1 ol <-> ul | **tlatilolca = tlatilulca** (*P022B*) |
| | 1 olo <-> olu | **xolotl = xolutl** (*P001A*) |
| | 1 om <-> um | **ompohualli = umpohualli** (*P022B*) |
| | 1 on <-> un | **oncan = uncan** (*P022B*) |
| | 1 ox <-> ux | **coxcoxtli = cuxcuxtli** (*3CHIMAL, P022B*) |
| | 1 teo <-> teu | **tlacateotzin = tlacateutzin** (*P022B*) |
| | 1 to <-> tu | **tollan = tullan** (*P022B, P311A*) |
| | 1 xo <-> xu | **xolotl = xulotl** (*P001A*) |
| | 2 cho <-> chu | **cholula = chulula** (*Citations Codex Xolotl*) |
| | 2 co <-> cu | **tencoatizque = tencuatizque** (*CF*) |
| | 2 coz <-> cuz | **tecozauitl = tecuzauitl** (*CF*) |
| | 2 eo <-> eu | **pineoac** = **pineuac** (*CF*) |
| | 2 io <-> iu | **michoacaiotl = michoacaiutl** (*CF*) |

[18] Les principes sont la synthèse des règles, ils permettent la comparaison avec les publications antérieures. Les règles sont celles qui sont effectivement utilisées par *GENOR*, mais sans leurs conditions. Le chiffre qui les précéde note le niveau : 1 pour les textes classiques, 2 pour ceux du *Codex de Florence* et 3 pour les *Techialoyan*. Le 0 sert à noter les quelques principes qui, bien que connus, ne sont pas appliqués. Les termes d'une règle sont unis par le signe <-> qui exprime leur caractère réversible. Cependant dans les quelques cas où ce caractère n'existe pas, on trouve le signe ->. Les textes, dans lesquels les exemples ont été trouvés, sont notés en abrégé. Soit par leur nom de fichier (*P312B*....) soit *CF* pour le *Codex de Florence*, soit *TECHIA* pour les textes *Techialoyan*.

[19] M. Launey. ; Th. Sullivan.

[20] M. Launey. ; Th. Sullivan.

[21] M. Launey.

[22] R. Andrews. ; M. Launey.

[23] R. Andrews. ; M. Launey. ; Th. Sullivan.

| | | | |
|---|---|---|---|
| | 2 | lo <-> lu | **elotl = elutl** (*CF*) |
| | 2 | mo <-> mu | **mochioa = muchioa** (*CF*) |
| | 2 | no <-> nu | **nochtli** = **nuchtli** (*CF*) |
| | 2 | ocpa <-> ucpa | **atocpan** = **tucpan** (*Citations Codex Xolotl*) |
| | 2 | otl <-> utl | **otli = utli** (*CF*) |
| | 2 | po <-> pu | **qualpol = qualpul** (*CF*) |
| | 2 | zo <-> zu | **auazoaio = auazuaio** (*CF*) |
| o <-> ou[24] | 1 | oa <-> oua | **epcoatzin = epcohuatzin** (*P022B*) |
| | 1 | oioa <-> oioua | **coioaque** = **coiohuaque** (*P022B, 303PBN*) |
| o <-> oo | 2 | oa <-> ooa | **mocuicuiloa = mocuicuilooa** (*CF*) |

## Voyelles et Consonnes

| Principes | Niveaux + Règles | | Exemples |
|---|---|---|---|
| ac <-> acu[25] | 0 | | **tlatzac(can) = tlatzacu(alli)** (*CF*) |
| acu <-> auc[26] | 1 | acu <-> auc | **hualmotzacuh = hualmotzauc** (*P022B, P312A*) |
| cn <-> cun | 1 | chicn <-> chicun | **chicnahui** = **chicunahui** (*P001A*) |
| cn <-> un | 1 | chicn <-> chiun | **chicnahui = chiuhnahui** (*P001A, P022B*) |
| ecu <-> euc[27] | 1 | ecu <-> euc | **tecuhtli = teuhctli** (*P022B, 3CHIMAL*) |
| li <-> ll[28] | 0 | | **pilyotl = pillotl** |
| mm <-> um[29] | 1 | ximm <-> xium | **toximmolpilli = toxiuhmolpilli** (*P312A*) |
| | 2 | uamm <-> uaum | **quammaitl = quauhmaitl** (*CF*) |
| nu <-> u[30] | 1 | anua <-> aua | **ititlanuan = ititlauan** (*P312A*) |
| pp <-> up[31] | 1 | app <-> aup | **nappohual = nauhpohual** (*P312A*) |
| zi <-> zz[32] | 2 | zio <-> zzo | **nequizio = nequizzo** |

[24] R. Andrews. ; M. Launey.
[25] M. Launey.
[26] M. Launey.
[27] M. Launey.
[28] Th. Sullivan.
[29] Th. Sullivan.
[30] M. Launey.
[31] Th. Sullivan.
[32] Th. Sullivan.

## Consonnes

| Principes | Niveaux + Règles | Exemples |
|---|---|---|
| b <-> p | 3 b <-> p | **sebastian = xepaxtian** (*TECHIA*) |
| c <-> q[33] | 1 cua <-> qua | **cuauhcihuatzin = quauhcihuatzin** (*3CHIMAL*) |
| | 3 cue <-> que | **mocuepa = moquepa** (*TECHIA*) |
| | 3 cui <-> qui | **macuil = maquil** (*TECHIA*) |
| c <-> qu[34] | 2 ic <-> iqu | **nicitta = niquitta** |
| c <-> x | 3 c <-> x | **oacico = oaxico** (*TECHIA*) |
| c <-> z | 1 ce <-> ze | **cempohualli** = **çempohualli** (*P022B*) |
| | 1 ci <-> zi | **citlaltepec = çitlaltepec** (*P022B*) |
| | 3 c <-> z | **cihuapili = zihuapili** = **sihuapili = çihu apili** (TECHIA) |
| ch -> h | 3 ch -> h | **chimalpopoca = himalpopoca** *;* **tlacochcalco = tlacohcalco** (*TECHIA*) |
| ch <-> x | 2 pich <-> pix | **Acamapichtli = Acamapixtli** (*Citations Codex Xolotl*) |
| ch <-> chch[35] | 2 ch <-> chch | **tlamachiuqui = tlamachchiuhqui** (CF) |
| ch <-> tzch[36] | 2 ch <-> tzch | **michiuh = mitzchiuh** (*CF*) |
| chtz <-> tz[37] | 2 tz <-> chtz | **nochpuchtzin = nochputzin** (*CF*) |
| d <-> t | 3 d <-> t | **mendoza = mentoza** (*TECHIA*) |
| l <-> ll[38] | 1 l <-> ll | **panquetzaliztli** = **panquetzalliztli** (*303PBN, 3CHIMAL*) ; **totoli** = **totolli** (*P022B*) ; **quimpiloque** = **quinpilloque** (*CASTILLO, P311A*) |
| m <-> mm[39] | 1 m <-> mm | **cemanahuac = cemmanahuac** (*CASTILLO*) |
| m <-> n[40] | 1 m <-> n | **impam** = **inpan** (*P311A*) |
| | 1 mp <-> np | **ipampa** = **ipanpa** (*3CHIMAL*) |
| | 1 mm <-> nm | **ommotlalli = onmotlalli** (*3CHIMAL*) |
| | 3 xam <-> xan | **xamicolax = xanicolax** (*TECHIA*) |
| n <-> nn[41] | 1 n <-> nn | **anozo = annozo** (*P311A*) |

[33] R. Andrews. ; M. Launey.
[34] R. Andrews. ; M. Launey.
[35] M. Launey.
[36] M. Launey.
[37] M. Launey.
[38] M. Launey.
[39] M. Launey.
[40] M. Launey. ; Th. Sullivan.
[41] M. Launey.

| | | | |
|---|---|---|---|
| n <-> Ø[42] | 1 | an -> a | **temazcaltitlan = temazcaltitla** (*P022B*) |
| | 1 | en <-> e | **cenyohual = ceyohual** (*P311A, P312B*) |
| | 1 | in -> i | **citlalin = citlali** (*P217A, P312A*) |
| | 1 | on -> o | **chiconxihuitl = chicoxihuitl** (*3CHIMAL, P312A*) |
| | 3 | men <-> me | **mentoza = metoza** (*TECHIA*) |
| | 3 | xan <-> xa | **xantiaco = xatiaco** (*TECHIA*) |
| p <-> pp | 1 | pa <-> ppa | **cepa = ceppa** (*3CHIMAL*) ; **ipan = ippan** (*3CHIMAL*) ; **inicopa = inicoppa** (*P022B, P311A*) |
| que <-> qe | 3 | que <-> qe | **altepehuaque = altepehuaqe** (*TECHIA*) |
| tz <-> tztz[43] | 2 | tz <-> tztz | **atzapotl = atztzapotl** (*CF*) |
| tz <-> z[44] | 1 | itz <-> iz | **panquetzalitztli = panquetzaliztli** (*CASTILLO, 303PBN*) |
| | 1 | quetz <-> quez | **nequequetzalo = inenequequezallo** (*P022B, 303PBN*) |
| | 1 | tetz <-> tez | **tetzcoca** = **tezcoca** (*3CHIMAL*) |
| | 3 | tz <-> z | **çentzontli = zenzontli** (*TECHIA*) |
| x <-> z | 3 | x <-> z | **axcapotzal = azcapotzalco** (*TECHIA*) |
| z <> zz[45] | 2 | z <-> zz | **ieço = iezzo** (*CF*) |

La lecture de ce tableau montre que les règles (au nombre de 81) sont nettement plus nombreuses et plus étoffées que les principes (39) et qu'un niveau leur est affecté. Ces deux caractéristiques répondent à un même souci qui est la limitation des formes générées. Même si le principe de *GENOR* est que peu importe le nombre de formes générées pourvu que toutes les graphies indispensables soient présentes et qu'il y ait un minimum de formes engendrant du bruit, il s'avère que le "peu importe" doit cependant être sérieusement contrôlé. En effet que se passerait-il si l'on appliquait les principes tels quels au mot **cuauhxilotl** ?

42 R. Andrews. ; M. Launey.; Th. Sullivan.
43 M. Launey.
44 R. Andrews.
45 M. Launey.

C c <-> q ; c <-> qu ; c <-> z
U o <-> u
A
U o <-> u
H
X c <-> x ; ch <-> x
I i <-> ii ; i <-> e
L l <-> ll
O o <-> u ; o <-> ou ; o <-> oo
T d <-> t
L l <-> ll

La mise en oeuvre de ces 16 règles créerait, au premier passage c'est à dire avant le déclenchement de nouvelles règles rendues activables du fait des nouvelles formes générées, 65.536 formes. Ceci n'est évidemment pas acceptable.

Pour limiter les formes deux mesures ont été prises. Les règles ont été réparties en trois niveaux. Par ailleurs les principes ne sont jamais appliqués tels quels mais sont transposés en règles conditionnelles.

### a) Les niveaux

Les trois niveaux découlent de l'observation de la répartition des règles selon les corpus étudiés. Alors qu'un grand nombre de variations orthographiques s'observent dans les trois groupes de documents certaines sont par contre spécifiques à l'un où à l'autre.

On n'observe ni dans les textes classiques, ni dans ceux du *Codex de Florence* des substitutions mettant en oeuvre les principes suivants : b <-> p ; c <-> q ; c <-> x ; c <-> z ; ch <-> h ; d <-> t ; m <-> n ; n <-> 0 ; que <-> qe ; tz <-> z ; x <-> z . Ils n'apparaissent que dans les textes *Techialoyan*.

De la même façon on n'observe pas dans les textes classiques des modifications mettant en oeuvre les principes suivants : e <-> ie ; o <-> oo ; mm <-> um ; zi <-> zz ; c <-> qu ; ch <-> x ; m <-> mm ; z <> zz. Ils n'apparaissent que dans les textes *Techialoyan* et ceux du *Codex de Florence*.

Les niveaux correspondent à des ensembles de règles qui sont applicables à des groupes de textes. Le niveau 3 (application des règles de niveau 1 + 2 + 3) correspond aux textes *Techialoyan*. Le niveau 2 (application des règles de niveau 1 + 2) correspond aux textes du *Codex de Florence*. Tandis

que le niveau 1 (application des seules règles de niveau 1) correspond aux textes classiques dont la définition ne peut être que négative. Il s'agit des textes qui n'appartiennent ni au *Codex de Florence*, ni aux *Techialoyan*.

Sur un total de 78 règles, 22 sont de niveau 2 et 13 de niveau 3. Cela signifie que l'utilisation du niveau 1 évite le déclenchement de près de la moitié des règles.

### b) Conditions d'application des règles de génération

L'introduction des niveaux ne fournit qu'une limite relative puisque lors de l'utilisation du niveau 3 toutes les règles peuvent être déclenchées. Il est une autre manière de brider la génération des graphies qui consiste à rechercher les strictes limites contextuelles dans lesquelles une règle peut être appliquée.

Supposons que l'on constate que **c** bien souvent remplace **q**. Il n'est malheureusement pas possible de poser la règle c -> q, sans autre forme de procès. Il faut en fait trouver la plus longue chaîne possible dans laquelle la règle c -> q est vrai. Dans les textes classiques la plus longue chaîne est **cua** ou **qua**. La règle est donc cua <-> qua. Toujours dans le même souci de ne pas permettre une génération débridée, il convient de s'assurer des effets de bord des chaînes constituant la règle. Ces effets indésirables sont (souvent mais pas toujours) écartés par l'utilisation de conditions négatives. Par exemple, dans les textes classiques, la règle teu <-> teo ne s'applique que si la lettre suivante de **teu** est différente d'un **i**. En effet dans ce cas on a affaire à des mots du type **teuilotl**, qui ne peuvent pas s'écrire ***teoilotl**. Pour connaître ces contextes, il faut d'une part trouver tous les mots comportant la chaîne **teu**, puis rechercher tous les mots comportant la chaîne **teo**. Transformer ensuite la chaîne **teu**, de la liste des mots en **teu**, en **teo** et éventuellement réduire les mots en supprimant affixes et suffixes. Prendre chacun des mots de cette nouvelle liste et rechercher s'il figure à l'intérieur de l'un des mots de la liste originelle des mots en **teo**. Supposons que l'on trouve **teucalli** dans la liste 1, on le transforme en **teocal**, puis on le lance en recherche sur la liste 2 qui comprend tous les mots du corpus comprenant la chaîne **teo**. Si l'on trouve le mot **teocalli**, on a la certitude que la règle teo <-> teu s'applique bien si la lettre suivant est un **c**. En pratiquant ainsi sur tous les mots de la liste, sachant que le corpus des textes classiques est constitué de 73 000 mots, celui du *Codex de Florence* de 53 622 mots et celui des textes *Techialoyan* de 29520 mots, cela permet de dresser une carte contextuelle relativement précise.

Une telle procédure de recherche contextuelle a été appliquée à chaque règle, mais autant elle peut donner de bons résultats sur des chaînes longues autant cela devient ingérable sur un petit nombre de caractères et particulièrement lorsqu'il s'agit d'un caractère unique. C'est bien pourquoi dans le cas de o <-> u le problème a été divisé en considérant ces lettres en association avec une consonne.

Dans le tableau qui suit on trouvera les principes, classés cette fois par niveaux, accompagnés des conditions qui en font des règles de génération. Quand un même principe est traité par plusieurs règles, les groupes de conditions se suivent séparés par un point-virgule[46].

Niveau 1

Voyelles

| | |
|---|---|
| e <-> i | si la lettre est précédée d'un **t** et suivi d'un **l**. |
| i -> ii | si **i** est suivi d'un **a**, d'un **o** ou d'un **e** et s'il n'est ni suivi ni précédé d'un **i**. |
| ii -> i | si **ii** est suivi d'un **a**, d'un **o** ou d'un **i** ou d'un blanc. |
| o -> u | si précédé de **c** et suivi de **io** ou de **co** ; si précédé de **e** et suivi de **a**; si suivi de **ca**, **ch** ou **m** ; si précédé de **ol** ou **te** ; si suivi de **l** et pas précédé de **l, h** ou **p** ; si suivi de **n** et pas précédé de **c** ou **n** ; si suivi de **x** et pas précédé de **a** ; si précédé de **t** et pas suivi de **u, h**, **a, o** ou un blanc ; si précédé de **x** et pas suivi de **u** ou **a** ; si précédé d'une voyelle et suivi d'une voyelle. |

46 Toutes les règles se trouvent dans un fichier nommé t2regle1.doc. Elles sont écrites selon un formalisme simple : _1_, _2_, _3_, _4_ dérivé du langage dBASE. 1 = chaîne d'origine ; 2 = chaîne transformée ; 3 = condition ; 4 = niveau. En langage naturel chaque règle correspond à une formule du type : transformer une chaîne "x" en une chaîne "y", si la condition est remplie et si son niveau est égal ou inférieur au niveau demandé.
La première règle exposée ici sous la forme e <-> i si la lettre est précédée d'un t et suivi d'un l est écrite :
_tel_, _til_, _1=1_, _1_
_til_, _tel_, _1=1_, _1_
La seconde double règle i -> ii si **i** est suivi d'un **a**, d'un **o** ou d'un **e** et s'il n'est ni suivi ni précédé d'un **i.**.
ii -> i si ii est suivi d'un **a**, d'un **o** ou d'un **i** ou d'un blanc, est écrite :.
_i_, _ii_, _substr(m2,pl-1,1)#"i" .and. substr(m2,pl+1,1)#"i" .and. (substr(m2,pl+1,1)="a" .and. (substr(m2,pl-1,1)=" " .or .substr(m2,pl+1,1)="o" .or .substr(m2,pl+1,1)="e")_, _1_

_ii_, _i_, _substr(m2,pl+2,1)="a" .or. substr(m2,pl+2,1)="o" .or. substr(m2,pl+1,1)=" " .or. substr(m2,pl+1,1)="i"_, _1_.
On peut voir que si les conditions dans le texte sont toujours indiquées d'une seule façon, en réalité elles peuvent être soit incluse dans la chaîne (cas de tel et til) ou bien traitée séparément, ce qui permet une plus grande souplesse.

u -> o — si précédé de **c** et suivi de **io** ou de **co** ; si précédé de **e** et suivi de **a** ; si suivi de **ca** ou **ch** ; si suivi de **m** et pas précédé de **a** ou **i** ; si précédé de **ol**, **t** ou **te** ; si suivi de **l** et pas précédé de **l**, **h** ou **p** ; si suivi de **n** et pas précédé de **c** ; si suivi de **x** et pas précédé de **a** ; si précédé de **x** et pas suivi de 'e ou **o** ; si précédé d'une voyelle et suivi d'une voyelle.

o -> ou — si le **o** est suivi d'un **a** et que la lettre précédente n'est pas **h**, **i**, **o**, **a**, **e** ou **u**.
ou -> o — si la lettre suivante est **a**.

## Voyelles et Consonnes

acu -> auc — si **acu** n'est pas précédé de **u** et est suivi ni de **l**, ni de **i**.
auc -> acu — si **auc** n'est pas suivi d'une voyelle.

cn <-> un — si **cn** est précédé de **chi**.

cun <-> cn — si **cun** est précédé de **chi**.

ecu -> euc — si la lettre précédente est différente de **u** et la lettre suivante différente de **h** ou de **l**.
euc -> ecu — si la lettre suivante n'est pas une voyelle.

mm <-> um — si **mm** ou **um** sont précédés par **xi**.

pp <-> up — si **pp** ou **up** sont précédés par **a**.

u <-> nu — si **u** ou **nu** sont précédés et suivis par **a**.

## Consonnes

c <-> q — si **c** ou **q** sont suivis de **ua**.

c <-> z — si **c** ou **z** sont suivis de **e** ou de **i** et dans ce dernier cas si la lettre précédente est différente de **x** ou de **z**.

l <-> ll — si **l** ou **ll** se trouvent entre deux voyelles.

m -> mm — si **m** se trouve entre deux voyelles.
mm -> m — dans tous les cas.

m -> n — si la lettre suivante est un blanc et que la deuxième lettre suivante est différente de **l** ; si la lettre suivante est un **p**.
n -> m — si la lettre suivante est un blanc et que la deuxième lettre suivante est différente de **l** et que la lettre précédente est différente de **i** ; si la lettre suivante est un **p**.

mm <-> nm — dans tous les cas.

mp <-> np — dans tous les cas.

n <-> nn — si **n** ou **nn** se trouvent entre les lettres **a**, **e**, **i** et **o**.

n -> Ø — si **n** est suivi d'un blanc et est précédé de **a**, **e**, **i** ou **o**.

p <-> pp — si **p** ou **pp** sont suivis de **a** et sont précédés de **e**, **i** ou **o**.

tz -> z — si **tz** est précédé de **te** ou de **que** ; si **tz** est précédé de **i** et qu'il n'est pas suivi d'une voyelle.

z -> tz si **z** est précédé de **te** ou de **que** ; si **z** est précédé de **i** et qu'il n'est pas suivi de **z**.

## Niveau 2

### Voyelles

e -> ie si le caractère précédent est un blanc.

o -> u si précédé de **ch**, **m** ou **n** ; si suivi de **cpa** ou **tl** ; si précédé de **c** et suivi de **z** ; si précédé de **z** et suivi de **l**, **c**, **z**, **q**, **i** ou **p** ; si précédé **c** et pas suivi de **u** ou **o** ; si précédé de **e** et pas suivi de **u** ou **i** ; si précédé de **z** et pas suivi de **u** ou **a** ; si précédé de **i** mais pas **li** et pas suivi de **a** ; si précédé de **l** mais pas **tl** et pas suivi de **u** ou **o**.

u -> o si précédé de **ch**, **m** ou **n** ; si suivi de **cpa** ou **tl** ; si précédé de **c** et suivi de **z** ; si précédé de **z** et suivi de **l**, **c**, **z**, **q**, **i** ou **p** ; si précédé **c** et pas suivi de **u**, **uu** ou **o** ; si précédé de **e** et pas suivi de **u** ou **i** ; si précédé de **z** et pas suivi de **u** ; si précédé de **i** mais pas **li** et pas suivi de **a** ; si précédé de **l** mais pas **tl** et pas suivi de **i**, **u** ou **o**.

o -> oo si la lettre suivante est un **a** et que le caractère précédent est différent de **h**, de **i**, de **o**, de **a**, de **e**, de **p** et de **u**.

oo -> o si la lettre suivante est un **a** et que le caractère précédent est différent de **h**.

### Voyelles et Consonnes

mm -> um dans tous les cas.

um -> mm si la lettre suivante est différente de **m** et de **n**.

zi <-> zz si **zi** ou **zz** sont suivis de **o**.

### Consonnes

c -> qu si la lettre précédente est **i** et si la lettre suivante est **i** ou **e**.

qu -> c si la lettre précédente est **i**.

ch -> x si les caractères précédents sont **pi**.

x -> ch si les caractères précédents sont **pi** et les suivants **t** ou un blanc.

z -> zz si les lettres précédentes sont **i** ou **e** et les suivantes **a**, **o**, ou **e**.

zz -> z dans tous les cas.

## Niveau 3

### Consonnes

b -> p dans tous les cas.

p -> b si la lettre précédente est **e** ou **a**.

c <-> q si **c** ou **q** sont suivis de **i** ou de **e** plus un blanc.

| | |
|---|---|
| c -> x | si la lettre précédente est une voyelle autre qu'un **u** et que la lettre suivante est **e** ou **i**. |
| x -> c | si la lettre précédente est une voyelle autre qu'un **u** et que la lettre suivante est **a**, **e** ou **i**. |
| c <-> z | si la lettre précédente est une voyelle, autre qu'un **u**, **c** ou un blanc et que la lettre suivante est **a**, **e**, **i** ou **p**. |
| ch -> h | si la lettre précédente est **a**, **o**, **e**, **n** ou un blanc. |
| d -> t | si la lettre suivante est différente de **t**. |
| t -> d | si la lettre précédente est **l** ou **n**. |
| m <-> n | si les lettres précédentes sont **xa**. |
| n <-> Ø | si les lettres suivantes sont **ent** ou **end**. |
| que <-> qe | si la lettre suivante est un blanc. |
| tz -> z | si la lettre précédente est différente de **z** et la suivante différente de **x** ou de **t**. |
| z -> tz | si la lettre précédente est différente de **t**, ou de **z** et la suivante différente de **z** ou de **t**. |
| x <-> z | si la lettre précédente est une voyelle, autre qu'un **u**, ou un blanc et la suivante une voyelle ou **c**, **t**, **q**, **m** ou encore un blanc. |

Tel est donc l'ensemble des conditions qui permettent de maintenir le processus de génération dans des limites raisonnables. Que donne l'application de ces règles conditionnelles aux deux mots choisis comme exemple ?

Pour **ihuan** *GENOR* crée, quelque soit le niveau utilisé, six formes : **IUAN, IUAM, IUA, IOAN, IOAM, IOA**.

Pour **cuauhxilotl**, au niveau 1, quatre graphies sont générées : **CUAUXILOTL, CUAUXILLOTL, QUAUXILOTL, QUAUXILLOTL**. Tandis que les niveaux 2 ou 3 génèrent huit orthographes : **CUAUXILOTL, CUAUXILUTL, CUAUXILLOTL, QUAUXILOTL, CUAUXILLUTL, QUAUXILLOTL, QUAUXILLUTL, QUAUXILUTL**.

A la suite sont reproduits les documents que crée automatiquement *GENOR* après chaque génération[47]. La colonne de gauche indique les graphies sur lesquelles les règles sont appliquées, la colonne centrale les règles avec leur niveau et la colonne de droite les graphies générées. La première ligne comporte

[47] Après chaque génération un fichier texte (genor_r.doc) est produit comportant la liste des formes générées avec pour chaque graphie la règle utilisée et son niveau.

à gauche le mot donné en entrée et à droite sa modification après application des règles de contraction et de transformation.

Niveau 1, 2 ou 3

| | | |
|---|---|---|
| **jhuan** | -> (0) | **IUAN** |
| **IUAN** | N->M (1) | **IUAM** |
| **IUAN** | AN->A (1) | **IUA** |
| **IUAN** | U->O (1) | **IOAN** |
| **IOAN** | N->M (1) | **IOAM** |
| **IOAN** | AN->A (1) | **IOA** |

Niveau 1

| | | |
|---|---|---|
| **cuauhxilotl** | -> (0) | **CUAUXILOTL** |
| **CUAUXILOTL** | L->LL (1) | **CUAUXILLOTL** |
| **CUAUXILOTL** | CUA->QUA (1) | **QUAUXILOTL** |
| **CUAUXILLOTL** | CUA->QUA (1) | **QUAUXILLOTL** |

Niveau 2 ou 3

| | | |
|---|---|---|
| **cuauhxilotl** | -> (0) | **CUAUXILOTL** |
| **CUAUXILOTL** | OTL->UTL (2) | **CUAUXILUTL** |
| **CUAUXILOTL** | L->LL (1) | **CUAUXILLOTL** |
| **CUAUXILOTL** | CUA->QUA (1) | **QUAUXILOTL** |
| **CUAUXILLOTL** | OTL->UTL (2) | **CUAUXILLUTL** |
| **CUAUXILLOTL** | CUA->QUA (1) | **QUAUXILLOTL** |
| **CUAUXILLUTL** | CUA->QUA (1) | **QUAUXILLUTL** |
| **CUAUXILUTL** | CUA->QUA (1) | **QUAUXILUTL** |

Au début il a été dit que le mot **cuauhxilotl** pouvait être orthographié de plusieurs milliers de façon différentes, on voit maintenant qu'il suffit de générer un maximum de huit formes pour pouvoir toutes les retrouver. Ceci n'est possible que par l'application, à différents stades du processus, des quatre catégories de règles : les règles d'expansion, de contraction, de transformation et de génération. De même les dix-neuf formes attestées du mot **ihuan** peuvent être retrouvées grâce aux six graphies créés par *GENOR*.

Ces exemples montrent comment *GENOR*, associé à *TEMOA*, vient à bout des orthographes variables, de la segmentation aléatoire et de la présence de signes paléographiques dans les textes nahuatl sur support informatique. L'antinomie relative entre le nahuatl et les ordinateurs est globalement réglée, il ne faudrait cependant pas croire que tout va pour le mieux dans le meilleur des mondes. *GENOR* présente un certain nombre de limites qu'il convient de mentionner.

- *GENOR* ne peut rien lorsque les règles n'ont pas encore été mises à jour !

- Les règles ne concernent que les documents étudiés jusqu'à maintenant. Cela exclut, en particulier, tous les textes modernes.

- Les quelques règles de niveau 0 ne sont pas appliquées. Généralement parce qu'elles sont hautement productives tout en s'appliquant à des cas marginaux.

- Certaines règles sont appliquées mais avec des conditions restrictives qui laissent peut-être de côté certains cas où l'application pourrait être nécessaire.

- Du fait de la progression géométrique du nombre de formes générées, *GENOR* ne peut s'appliquer, en particulier avec le niveau 3, à des mots trop longs.

- Certaines transformations, celles qui s'éloignent trop du référent phonétique, ne peuvent pas être générées et donc trouvées alors qu'une lecture directe permettrait de les reconnaître. Le couple oeil/cerveau possède une souplesse que *GENOR* ne peut égaler. Certaines orthographes sont totalement imprévisibles. On le voit par exemple chez Ixtlilxochitl qui écrit **Chicnauhtlan Ziauhthnauhtlan**[48], et **Motezulmaltzin** pour **Motecuhzomatzin**[49]. De plus certaines de ces orthographes déroutantes sont volontaires tandis que d'autres peuvent simplement correspondre à des erreurs d'un copiste. Ainsi dans la *Crónica Mexicayotl*[50] le mot **Chalchiuhnenetzin** est écrit **Chialchiuhnenetzin**. Quelle que soit l'explication donnée aux graphies inattendues, celles-ci peuvent occasionner des dysfonctionnements préjudiciables à la recherche.

- Il y a des générations qui ne sont pas toujours souhaitées mais qui ne sont pas évitables, soit les deux règles cua <-> qua et co <-> cu, et le mot **coatl** à générer. L'application de co <-> cu va donner **cuatl**, puis sur cette forme l'application de cua <-> qua va donner **quatl**. Terme indésirable, surtout si l'on travaille sur les racines. Pour éviter cette génération il faudrait limiter l'application de co <-> cu, mais cela n'est pas possible car il existe des mots comme **tencoatizque = tencuatizque** dans le *Codex de Florence*.

---

48 IXTLILXOCHITL, Fernando de. 1975-77. *Obras Históricas*, Edición... por Edmundo O'Gorman, México, UNAM, Tome I 566 p., Tome II 565 p. Relaciones I, p. 325

49 *Relaciones* I, p. 321

50 page 107 ou folio 52

- Toutes les procédures informatiques mises en oeuvre pour la détermination des règles, l'assignation d'un niveau et la recherche des contextes d'application souffrent d'un même défaut. Elles sont très efficaces pour toutes les chaînes significatives (généralement longues) et guère pour les autres obligeant à des contrôles manuels aléatoires. Les flexions, le caractère agglutinant du nahuatl, la segmentation irrégulière et le fait que les variations orthographiques peuvent affecter diverses parties d'un mot rendent difficiles tous les contrôles automatiques.

Les limites exposées ci-dessus n'ont absolument pas un caractère inexorable. On peut au contraire penser que certaines pourront être repoussées voir supprimées. Ceci est d'autant plus aisé que *GENOR* étant un système expert ses règles sont modifiables à tout instant, par tout un chacun, puisqu'il est simplement nécessaire d'utiliser un traitement de texte et de se conformer au formalisme des expressions[51].

[51] Les règles de *GENOR* se trouvent dans un fichier nommé t2regle1.doc qui comprend au début toutes les explications nécessaires pour modifier, supprimer ou ajouter des règles.