# H : un langage de description des données historiques. Application au traitement des données de l'état civil ancien

Jean-Claude Poupa

# H

## Un langage de description des données historiques
## *Application au traitement des données de l'état civil ancien*

**Jean-Claude Poupa**[1]
INRA
INSTITUT NATIONAL DE LA RECHERCHE AGRONOMIQUE
Unité d'économie et sociologie rurales
65 rue de Saint-Brieuc, 35042 Rennes cedex.

2 juillet 1998

[1]e-mail : jcpoupa@roazhon.inra.fr

Une forme classique de l'information historique est un document écrit, élaboré par une autorité officielle dans le contexte politique de l'époque. Le texte du document décrit **un évènement survenu en un jour en un lieu**, lequel mobilise une ou plusieurs personnes avec des rôles dûment établis. La description de l'évènement s'effectue selon des protocoles qui demeurent proches sur de longues périodes : actes de l'état civil, minutes des actes notariés, rôles des services fiscaux, émargement dans un registre, etc. Ces documents sont dans la suite appelés **actes**. Un ensemble d'actes de même nature est un **registre**.

Le dépouillement des documents historiques apparaît traditionnellement comme une activité qui se prête mal à la formalisation mathématique. Or la puissance de calcul logique des ordinateurs fait que certains instruments mathématiques deviennent aujourd'hui efficaces pour réaliser des traitements de grande envergure jusqu'alors inenvisageables. C'est dans le but d'apporter à l'historien ces méthodes relevant des mathématiques appliquées que nous avons réalisé, dans le cadre du programme de recherche PAGI sur *la mobilité géographique et sociale des populations anciennes*[1], une première modélisation, laquelle a débouché sur une spécification formelle et la réalisation d'un prototype logiciel opérationnel, utilisable pour traiter les données de l'état civil.

## Une modélisation basée sur la théorie des langages formels

Dans une chronologie historique, l'évènement est l'action qui initialise ou modifie la valeur d'une variable d'état d'un processus relatif à une personne : naissance, mariage, divorce, décès, acquisition d'un bien, incorporation militaire, inscription sur une liste électorale, etc.

Les actes sont, selon leur nature, structurés selon des modèles relativement stables sur des périodes historiques, conformément à des règles et usages qui répondent aux exigences juridiques, politiques ou religieuses d'une époque. Ainsi, les variables principales de l'état civil laïc d'après la Révolution sont les mêmes que celles antérieurement utilisées pour les actes de mariages, baptêmes et sépultures établis par les autorités religieuses.

Une personne citée est identifiée dans un acte par un nom et un ou plusieurs prénoms, transcrits en appliquant des règles syntaxiques et orthographiques : cet identifiant civil constitue une valeur textuelle **atomique**

[1]**Jean-Pierre Pélissier**, INRA, Économie et Sociologie Rurales, 65 boulevard de Brandebonrg, 94205 Ivry-sur-Seine cedex.

gérée dans une **variable historique.** La date de naissance de cette personne exprimée par un jour, un mois et une année dans un calendrier est une autre valeur atomique usuellement représentée par une suite ordonnée de trois nombres, gérée dans une autre variable. La valeur atomique du lieu de naissance est une hiérarchie de toponyme calquée sur une organisation administrative. Une suite de variables est ainsi définie dans un **espace historique** afin de pouvoir décrire les personnes citées dans les actes. La valeur d'une variable, usuellement appelée **observation**, traduit un état de la personne à la date d'établissement de l'acte dans lequel elle est citée.

Le langage $H$ formalise une représentation de ces valeurs historiques extraites de documents d'archives. C'est un *langage formel* défini par une *grammaire algébrique* et reconnu par un *automate.*[2]

### Un compilateur pour automatiser les contrôles syntaxiques et normaliser l'écriture

L'intérêt d'un langage formel est de spécifier un algorithme traduit par un compilateur puis exécuté par une machine pour détecter des erreurs et normaliser la représentation de l'information.

Un acte est un texte structuré duquel il est possible d'extraire des valeurs de variables historiques définies sur des ensembles appelés **domaines**, et regroupées dans des **classes** :

les patronymes en usage sont énumérés dans un annuaire ;

Les dates calendaires d'une période forment un ensemble fini ;

les toponymes officiels sont répertoriés dans un dictionnaire ;

les métiers sont regroupés dans des lexiques.

La grammaire du langage $H$ définit un ensemble de mots noté $\mathcal{H}$.
Les valeurs de $m$ variables lues dans un acte définissent une observation notée $\mathcal{H}^m$.
Un registre de $n$ actes est formellement un élément de $\mathcal{H}^{nm}$ qui peut être visualisé dans un tableau à $n$ lignes et $m$ colonnes : $h_{ij} \in \mathcal{H}$ désigne la valeur

[2] La théorie des langages formels constitue l'un des modèles de calcul les plus anciens en informatique. Pour une première approche, on pourra consulter le cours de **Danièle Beauquier**, **Jean Berstel** et **Philippe Chrétienne**, publié aux éditions Masson en 1992 sous le titre *Éléments d'algorithmique.*

de la variable $j$ de l'acte $i$.
Une classe est un sous-ensemble de $\mathcal{H}$.

Les instructions de traitement des registres sont codées dans un document de paramètres, appelé **pilote**. Elles permettent successivement, pour un ensemble d'actes de même nature :

- le calcul des rôles des personnes citées,
- la répartition des variables dans les classes,
- la description des différents types de variables,
- l'interprétation des valeurs textuelles en fonction de la forme syntaxique d'acquisition,
- l'assemblage des composants syntaxiques élémentaires.

## Un modèle de données relationnel pour archiver les valeurs des variables historiques

Les valeurs des variables historiques sont gérées dans des **relations**.[3] Le *calcul relationnel* est ensuite utilisable pour effectuer des traitements : contrôles de cohérence au moyen des opérateurs ensemblistes, automatisation de la correction des erreurs fréquentes, production de tableaux de bord, archivage, regénération des actes après apuration, ...

Les variables des actes d'un registre sont lues consécutivement, analysées syntaxiquement de façon globale et dans les classes, puis transcrites sous une forme normalisée en l'absence d'erreur ou ambiguïté. Les valeurs sont identifiées par une clé numérique et gérées dans un **fonds d'archivage**.

Les **dictionnaires** sont vus comme des actes contenant une seule variable : un registre est alors une liste de valeurs. En conséquence, les mots d'un dictionnaire sont contrôlés et normalisés de la même façon que ceux lus dans les actes, ce qui garantit l'efficacité des opérateurs d'appartenance ensembliste.

[3] La théorie de l'algèbre relationnelle, construite à partir du concept mathématique de relation n-aire, constitue un modèle de données simple et robuste. L'ouvrage de **Claude Delobel**, **Christophe Lécluse** et **Philippe Richard** présente les concepts essentiels en algorithmique des données dans un ouvrage intitulé *bases de données : des systèmes relationnels aux systèmes à objets* publié chez INTEREDITIONS en 1991.

### Des liens entre actes modélisés dans la théorie des graphes

Certaines variables lues dans un acte gèrent des états relatifs à des personnes et consécutifs à des faits pour lesquels des actes ont normalement été établis : dates et lieux de naissance ou décès, mentions marginales pour l'état civil, reconnaissances d'enfants, indications selon lesquelles des personnes sont mariées, veuves, vivantes, décédées...

Ces situations définissent des liens entre les actes qui peuvent être représentés dans un graphe : les sommets sont les actes et l'existence d'un arc orienté indique qu'une personne citée dans l'acte associé à l'origine, dit source, est sujet dans l'acte associé à l'extrémité, dit cible.

La nature d'un lien est reconnne par le compilateur qui détermine les rôles de la personne dans les actes concernés et produit une clé d'identification de ce lien appelée **référence** et gérée dans nne relation.

Les algorithmes classiques de cheminement sont utilisables pour réaliser des contrôles du type absence de boucles, automatiser des recherches, isoler des composantes connexes, ... [4] Nous nous limitons ici à la construction des graphes au moment de la reconnaissance des données afin de les stocker en vue de traitements ultérieurs.

[4] Les algorithmes utilisés dans la théorie des graphes sont présentés dans l'ouvrage de **Michel Gondran** et **Michel Minoux** intitulé *Graphes et algorithmes*, publié aux éditions EYROLLES en 1985.

# Chapitre 1

# Définition des espaces historiques

L'objectif de cette modélisation est globalement d'aboutir à une numérisation des données factuelles historiques pour pouvoir réaliser les calculs sur une algèbre de nombres. Celà revient à définir un **isomorphisme** entre un ensemble $\mathcal{F}$ de faits définis sur un espace, dit historique, et l'ensemble $\mathcal{N}$ des entiers naturels. A l'issue de ces calculs, l'information devra évidemment pouvoir être restituée sous sa forme initiale.

Ce chapitre définit d'un point de vue formel ces espaces historiques. Il décrit la grammaire du langage $H$ qui permet une formalisation des représentations textuelles des faits dans une langue nationale et présente les instruments algébriques utilisés pour effectuer cette modélisation.

## 1.1 Syntaxe du langage H

Le problème à résoudre au vu des données factuelles lues dans les actes est celui de la définition de l'instrument formel efficace pour gérer tous les cas de figures sans perte d'informations.

Pour ce faire, des structures syntaxiques minimales ont été définies afin de pouvoir gérer en pratique des noms propres et communs, des nombres, des listes et des précisions placées entre parenthèses : des exemples sont fournis plus loin au paragraphe 1.1.3.

### 1.1.1 Notations

Les grammaires sont décrites avec les conventions suivantes :

$< lettre >$ représente une **variable syntaxique**,
$\longrightarrow$ est l'opérateur de dérivation,
$a$ représente un élément terminal de l'alphabet ,
la barre verticale | traduit un choix entre plusieurs dérivations,
$[< lettre >]$ traduit zéro ou une occurrence de $< lettre >$,
$\{< lettre >\}$ signifie zéro ou plusieurs occurrences de $< lettre >$.

Ce formalisme est utilisé pour la grammaire générale et pour les sous-ensembles propres aux classes afin de décrire sans ambiguïté les formats utilisés dans l'environnement PAGI. Les mots de l'ensemble $\mathcal{H}$ sont définis à partir d'un alphabet $\mathcal{A}$ et de l'*axiome* $< citation >$.

Des restrictions syntaxiques supplémentaires sont introduites sur cette syntaxe pour définir une forme normale. D'autres contraintes syntaxiques sont introduites dans les classes sur ces valeurs textuelles normalisées.

### 1.1.2 Grammaire générale

L'objectif de cette grammaire est de formaliser la représentation des faits élémentaires, décrits par des variables d'états dont les valeurs sont construites par assemblage de **noms**.

Les noms sont définis comme des *mots ou groupes de mots servant à désigner un individu et à le distinguer des êtres de la même espèce.*[1] Un nom peut donc être constitué de plusieurs mots séparés par des espaces.

Le mode d'assemblage des noms exprime un lien entre les noms, pratiquement une séquence ou une inclusion.

#### Dérivations de l'axiome

$$
\begin{aligned}
< citation > \longrightarrow\ & < terme >\{< suite >\} \mid \\
& < entete > \{< suite >\} \mid \\
& < entete > [(< inclusion >)[< espace >]] \mid \\
& < vide >
\end{aligned}
$$

La variable $< entete >$ a été introduite pour pouvoir répéter des données globales mais en restreignant cette possibilité au premier élément d'une liste. La variable $< vide >$ permet de coder l'absence explicite d'information.

[1] Définition du Petit Robert 1.

### Dérivations intermédiaires

$< suite > \longrightarrow < delimiteur >< terme >$
$< terme > \longrightarrow < groupe > [(< inclusion >)[< espace >]]$
$< groupe > \longrightarrow < champ >\{< inter >< champ >\}$
$< inclusion > \longrightarrow < groupe >\{< delimiteur >< groupe >\}$
$< champ > \longrightarrow [< espace >] < mot > [< espace >]$
$< inter > \longrightarrow < espace > | < separateur >$
$< espace > \longrightarrow < blanc >\{< blanc >\}$
$< entete > \longrightarrow [< espace >] < repetiteur > [< mot >]$
$< mot > \longrightarrow < nom > | < nombre > | < alphanu >$
$< nom > \longrightarrow < lettre >\{< lettre >\}$
$< nombre > \longrightarrow < chiffre >\{< chiffre >\}$
$< alphanu > \longrightarrow < nom >< nombre > | < nombre >< nom >$
$< lettre > \longrightarrow < latin > | < joker >$
$< chiffre > \longrightarrow < arabe > | < joker >$

Ces règles permettent pratiquement de gérer des listes de termes délimités par le composant syntaxique $< delimiteur >$. Un terme est un groupe nominal qui peut être suivi d'une expression parenthèsée, laquelle contient une liste de groupes nominaux. Cette construction n'autorise qu'un seul niveau de parenthèsage à droite d'une chaîne non vide.

### Dérivations terminales

$< delimiteur > \longrightarrow ,|;|:$
$< separateur > \longrightarrow -|'|/$
$< vide > \longrightarrow \epsilon$[2]
$< repetiteur > \longrightarrow =$
$< blanc > \longrightarrow \_$[3]
$< latin > \longrightarrow a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|$
$\quad ç|à|è|é|ù|â|ê|î|\ ô|û|ä|ë|ï|ö|ü|ÿ|$
$\quad A|B|C|D|E|F|G|H|I|J|K|L|M$
$\quad N|O|P|Q|R|S|T|U|V|W|X|Y|Z$
$< arabe > \longrightarrow 1|2|3|4|5|6|7|8|9|0$
$< joker > \longrightarrow .|?|*$

[2] $\epsilon$ représente la chaîne vide.

[3] Le tiret souligné(_) matérialise ici l'espace séparateur mais n'appartient pas à l'alphabet donc est interdit dans la transcription des actes.

Cette grammaire est définie sur l'alphabet français. L'alphabet pourrait être étendu aux majuscules accentuées Â, È, É, Ô. Il est possible de choisir un autre alphabet : pour la langue espagnole, par exemple, on supprimera certaines lettres accentuées et on introduira *í*, *ó*, *ñ*.

L'alphabet et les transitions sont des constantes propres à une langue nationale : les fonctions de passage entre les formes minuscules et majuscules sont redéfinies explicitement afin de pouvoir restituer si nécessaire les orthographes conventionnelles propres aux classes.

#### Compilation

Le langage $H$ est un *langage rationnel* au sens de la *théorie des langages formels.*[4] Le *théorème de Kleene* permet de conclure à l'existence d'un *automate d'états finis*, lequel définit un algorithme de reconnaissance des mots du langage $H$, qu'il suffit ensuite de traduire pour produire un compilateur.

### 1.1.3 Exemples d'utilisation

L'objectif est de gérer les noms et les dates, isolés, énumérés ou hiérarchisés. Les mots sont construites par concaténation d'autres mots. Le mot vide $\epsilon$, élément neutre de l'ensemble $\mathcal{H}$ pour l'opération de concaténation, code habituellement l'absence d'information pour la variable, que celle-ci soit sans objet ou non transcrite.

Ces fonctionnalités pratiques sont illustrées avec les valeurs suivantes placées entre guillemets :

- « *Tiphaine Cancouët;Jean-Pierre Pélissier;Jean-Claude Poupa* » représente une liste de noms avec le point-virgule comme délimiteur.
- « *Poupa(dit Cadet),Célestin Isidore* » est une transcription des noms de personnes dans laquelle
  - la virgule délimite le patronyme et les prénoms,
  - l'espace sépare les prénoms successifs,
  - les parenthèses introduisent une précision relative au patronyme.

[4] La théorie linguistique des *langages à états finis* élaborée par **Noam Chomsky** en 1956 a fait l'objet d'un ouvrage dont une traduction est disponible dans la collection Points des Éditions du Seuil sous le titre *Structures syntaxiques.*

- Le choix du séparateur permet de codifier des règles ou des usages :
  - le tiret fait de « *Marie-Magdeleine* » un prénom composé ;
  - l'espace fait de « *Marie Magdeleine* » une liste de deux prénoms ;
  - la barre oblique de « *Sully/Loire* » remplace la préposition *sur* ;
  - l'apostrophe traduit l'élision d'une voyelle dans « *L'Isle-d'Abeau* », et plus rarement une spécificité phonétique régionale comme dans « *Penmarc'h* ».
- « *Coullons(Les Gaux);Dampierre(Ponta);Nevoy(Le Marais)* » est une liste de lieux avec le nom du hameau indiqué entre parenthèses derrière celui de la commune.
- « *20/02/1998* », « *20-02-1998* » , « *20 2 1998* » « *20 février 1998* » et « *25 floréal an08* » sont des dates.
- Les points dans « *J.* » et « *M.* » indiquent que la valeur est une abréviation, respectivement pour *Jean* et *Marie* dans cette application.
- Le point d'interrogation de « *Dupon?* » remplace une lettre inconnue.
- L'étoile de « *Dampierre-en-B*y* » remplace des lettres inconnue.
- Les parenthèses apportent une précision relative à une information éventuellement indéterminée mais non vide : « **(Levran)* » code un hameau d'une commune inconnue et « **(*)* » peut indiquer que le nom du hameau est de surcroît illisible.
- « *=H,Pierre* » signale que le patronyme de la personne est le même que celui du mari dans un acte de mariage.
- « *=(Les Telliers)* » signifie que le hameau cité est situé dans la commune d'établissement de l'acte.

Les deux derniers exemples illustrent le mode d'utilisation de la marque de répétition pour éviter la saisie multiple de noms propres figurant dans l'entête de l'acte, dès lors que l'orthographe est identique.

### 1.1.4 Normalisation syntaxique

Les mots reconnus sont des noms propres ou communs, des nombres décimaux et des codes. La normalisation consiste à définir des règles précises de gestion des espaces séparateurs. L'application de ces règles produit une forme syntaxique dite normale.

#### Génération de la forme normale

Les mots normalisés définissent un sous-ensemble de $\mathcal{H}$ généré en remplaçant la variable syntaxique $< terme >$ par $< hterme >$ qui se dérive alors comme suit :

$< hterme > \longrightarrow < hgroupe > [(< hinclusion >)]$
$< hinclusion > \longrightarrow < hgroupe > \{< delimiteur >< hgroupe >\}$
$< hgroupe > \longrightarrow < hchamp > \{< hsepa >< hchamp >\}$
$< hchamp > \longrightarrow < hnom > | < nombre > | < alphanu >$
$< hsepa > \longrightarrow < blanc > | < separateur >$
$< hnom > \longrightarrow < nom > \{< hsepa >< nom >\}$

#### Domaines de définition

Les mots dérivés du composant $< hnom >$ sont les noms et groupes de noms d'une langue nationale : ils définissent un premier ensemble $\mathcal{G} \subset \mathcal{H}$ dont les éléments sont les entrées d'un dictionnaire de groupes nominaux.

Les expressions alphanumériques dérivées de $< hchamp >$, définissent un second ensemble $\mathcal{A} \subset \mathcal{H}$. C'est une extension de l'ensemble $\mathcal{G}$ qui autorise une utilisation partielle des chiffres : $\mathcal{G} \subset \mathcal{A}$.

## 1.2 Représentation algébrique

Les valeurs des variables historiques sont des séquences de texte plus ou moins structurées, du simple code numérique à la liste de termes complexe. La façon globale de gérer ces variables est indépendante de leur signification. Dans la représentation tabulaire classique, la valeur de la la $j^{ème}$ variable du $i^{ème}$ acte $A_i$ est notée $h_{ij}$

### 1.2.1 Représentation relationnelle

L'algèbre relationnelle permet de modéliser efficacement ces représentations. La description des relations et de leurs propriétés fondamentales nécessite l'introduction de notations particulières.

#### Schémas des relations

Les noms des relations sont notés en caractères gras, suivis de la liste des attributs entre parenthèses. Les attributs préfixés par la lettre $h$ sont

définis sur le domaine $\mathcal{H}$, les autres sur l'ensemble $\mathcal{N}$ des entiers naturels. Les attributs de la clé primaire sont notés en gras.

La relation $\mathbf{F(i,j},h)$ est suffisante pour gérer l'ensemble des valeurs habituellement représentées sous la forme tabulaire : l'unicité de la clé primaire traduit l'unicité de la valeur $h_{ij}$ de la variable $j$ dans l'acte $i$ .

#### Valeurs des relations

Les relations gèrent des ensembles dont les valeurs varient dans le temps. Ces ensembles sont notés avec une lettre calligraphique indicée pour préciser la période associée : $\mathcal{F}_t$ désigne la valeur de la relation $F$ à l'instant $t$, soit l'ensemble des faits $h_{ij}$ répertoriés à cet instant.

Les éléments d'un ensemble géré dans une relation sont des n-uplets, appelés tuples. Les valeurs des tuples sont représentées avec les lettres grecques : le triplet $(\alpha, \mu, \varphi)$ désigne le fait $\varphi$ décrit par la variable $\mu$ pour un acte $\alpha$.

### 1.2.2 Ordonnancement

La clé primaire d'une relation est par définition suffisante pour identifier tout élément de l'ensemble contenu dans cette relation mais elle ne permet pas de classer les éléments de cet ensemble. Or les calculs sur de grands ensembles, qu'ils soient effectués manuellement à travers une stratégie de rangement ou par un algorithme de tri exécuté par une machine, nécessitent des ordonnancements préalables pour être efficaces. Les relations d'ordre associées sont à conserver, ce que ne permet pas le modèle relationnel dans sa définition théorique, d'où la nécessité de compléter cette modélisation.

#### Définition d'une relation d'ordre total

Soit la relation $\mathbf{R(a_1},...,\mathbf{a_p},h_{p+1},...,h_q)$ et l'ensemble $\mathcal{R}_t$ des tuples de $R$ à l'instant $t$, avec $a_1,...,a_p \in \mathcal{N}$ et $h_{p+1},...,h_q \in \mathcal{H}$.

L'unicité de la clé primaire d'une relation implique que

$$\forall(\alpha_1,...,\alpha_i,...,\alpha_p) \in \mathcal{R}_t, \forall(\beta_1,...,\beta_i,...,\beta_p) \in \mathcal{R}_t,$$
$$\exists k \leq p : \forall i < k, (\alpha_i = \beta_i) \wedge (\alpha_k \neq \beta_k).$$

On en déduit l'ordonnancement suivant :

si $(\alpha_k < \beta_k)$ alors $(\alpha_1,...,\alpha_i,...,\alpha_p) < (\beta_1,...,\beta_i,...,\beta_p)$
sinon $(\beta_1,...,\beta_i,...,\beta_p) < (\alpha_1,...,\alpha_i,...,\alpha_p)$.

### Choix des relations d'ordre et notations

Des ordonnancements peuvent être définis pour toute permutation des attributs d'une clé primaire dans cette notation de gauche à droite. Le point-virgule est utilisé dans les notations suivantes pour traduire explicitement la séquence retenue dans l'ordonnancement désigné. La relation d'ordre choisie est repérée par une valeur d'indice.

Ces notations sont illustrées sur l'exemple d'un dictionnaire des communes françaises géré dans la relation

$$\mathbf{D}(\mathbf{departement}, \mathbf{commune}, \mathit{region}, \mathit{toponyme}).$$

L'ordonnancement noté

$$\mathbf{D_1}(\mathbf{departement}; \mathbf{commune}; \mathit{region}, \mathit{toponyme})$$

place en tête la clé $(1, 1, 22)$ qui identifie la commune de *L'Abergement-Clémenciat* dans le département de l'*Ain* de la région *Rhône-Alpes*. En revanche, si le code de la région est le rang alphabétique, l'ordonnancement

$$\mathbf{D_2}(\mathbf{region}; \mathbf{departement}; \mathbf{commune}; \mathit{toponyme})$$

place en premier la clé $(1, 67, 1)$ qui désigne la commune d'*Achenheim* dans le département du *Haut-Rhin* de la région *Alsace*.

La valeur d'une relation étant définie à l'instant $t$, les ensembles $\mathcal{D}_{1t}$ et $\mathcal{D}_{2t}$ représentent à cet instant les valeurs de deux dictionnaires classés différemment pour un même ensemble $\mathcal{D}_t$.

### Optimisation des calculs relationnels

Cette utilisation des **relations d'ordre** binaires[5] pour classer les tuples d'une relation est un moyen de signaler au « moteur relationnel » l'existence d'ordonnancements de telle sorte qu'il puisse mettre en œuvre des algorithmes adaptés aux calculs sur de grands ensembles.

## 1.2.3 Définition des équivalences

La comparaison des noms est une opération complexe qui est abordée ici seulement pour introduire les méthodes formelles de construction. Nous reprenons pour celà la définition mathématique de la relation d'équivalence.

[5] Le terme *relation* est utilisé pour désigner à la fois une relation binaire quelconque et la relation n-aire $R$ dont la valeur $\mathcal{R}_t$ à l'instant $t$ est l'ensemble sur lequel est défini cette relation binaire.

Soit

$\equiv$ un opérateur,
$x, y, z \in \mathcal{H}$ des opérandes quelconques,
$\varphi$ une fonction définie de $\mathcal{H}$ vers $\mathcal{N}$.

$$x \equiv y \text{ est vrai si } \varphi(x) = \varphi(y), \text{ faux sinon.}$$

$\varphi$ définit une relation d'équivalence. En effet

1. $x \equiv x$ (réflexivité) ;
2. $x \equiv y \Rightarrow y \equiv x$ (symétrie) ;
3. $(x \equiv y) \wedge (y \equiv z) \Rightarrow x \equiv z$ (transitivité).

### Comparaison simple des chaînes de caractères

L'opérateur le plus simple compare deux chaînes caractère à caractère: $x \equiv y$ est vrai seulement si les chaînes sont de même longueur et tous les caractères de même rang sont identiques, faux sinon. C'est l'opérateur par défaut associé au symbole = des langages d'interrogation des bases de données.

L'efficacité pratique d'un tel opérateur est améliorée d'abord par la normalisation syntaxique du paragraphe 1.1.4 qui supprime les espaces inutiles puis par l'application des règles orthographiques définies plus loin pour les différentes classes de variables.

En l'absence de règles strictes d'orthographe, pour les documents anciens en particulier, cet opérateur est peu efficace.

### Comparaison phonétique

Il est toujours possible d'énumérer des règles d'équivalence phonétique entre graphies : consonnes muettes, syllabes équivalentes, doublement de consomnes, représentation, écriture des voyelles nasales, orales, ...

Des listes de règles dûment formalisées sont utilisables pour construire des opérateurs de comparaison plus efficaces que le précédent pour calculer les vraies équivalences mais qui risquent d'en introduire de fausses. Un opérateur de ce type a été développé à titre expérimental.[6]

[6] **Olivier Le Medec**, *Construction d'un automate de reconnaissance de liens généalogiques Traitement de la variabilité orthographique des patronymes*, rapport de stage DESS informatique, INRA, IFSIC, 1994.

Cette méthode, utilisable pour rapprocher des noms, s'avère toutefois peu efficace pour automatiser le calcul des équivalences.

#### Gestion relationnelle de relations d'équivalences

La comparaison phonétique établit des listes d'équivalences possibles. Les usages en matière de transcriptions varient par ailleurs dans l'espace et le temps et dépendent du contexte social : le *Méen* de Saint-Méen en Ille-et-Vilaine et le *Meung* de Meung-sur-Loire dans le Loiret se prononcent de la même façon ; *Le Bressac* et *Le Bersac* sont deux écritures en usage d'un même hameau, le premier à caractère officiel et le second local.

Ces équivalences sont gérées de façon simple dans la relation

$$\mathbf{E}(\mathbf{hnom}, \mathbf{departement}, \mathit{classe}).$$

L'attribut numérique *classe* code la classe d'équivalence d'un nom en usage dans un département et géré dans l'attribut *hnom*. Ainsi, l'équivalence liée à l'inversion du *e* et du *r* dans le patronyme *Breton* qui se transforme en *Berton* dans certaines régions fait que ces patronymes appartiennent à la même classe d'équivalence pour les départements de la région.

Ce type de construction peut être affiné jusqu'au niveau de la commune ou pour des périodes par ajout d'attributs supplémentaires intégrés dans la clé primaire.

#### Génération des équivalences

Des classes d'équivalence simples sont parfois définies dans une variable contenant un plan de classement, par exemple la nomenclature des catégories socio-professionnelles. Les toponymes et patronymes sont en revanche gérés dans des ensembles complexes dont la cardinalité peut atteindre le million d'éléments : une codification exclusivement manuelle est donc exclue.

## 1.3 Identification des observations

La représentation algébrique définit l'observation élémentaire comme un élément de l'ensemble $\mathcal{H}$. Les attributs utilisés pour identifier ces observations, définis sur $\mathcal{N}$, sont préfixés par la lettre $i$.

Une observation est dans cette représentation relative à une personne citée de par son rôle dans un acte établi consécutivement à un évènement. Les

actes, regroupés dans des registres, sont globalement structurés en plusieurs parties :

- l'entête contient des données affectées aux sujets et utilisées pour classer les actes ;
- le corps regroupe les informations relatives aux personnes citées ;
- la marge contient des inscriptions ajoutées postérieurement à la date d'établissemeut de l'acte et concernant les sujets ;
- les notes, rassemblent des informations atypiques figurant dans l'acte ainsi que les commentaires éventuels ajoutés au vu de l'acte.[7]

### 1.3.1 Origine des actes

Les actes de même nature sont regroupés dans des ensembles homogènes repérés par des codes numériques gérés dans la variable *source*, définie comme la somme des trois variables

$$source = evenement + origine + lot.$$

La variable *evenement* de la table 1.1 code le fait juridique.
La variable *origine* du tableau 1.2 décrit la nature du document.
La variable $lot \in [0...999]$ contient un numéro d'ordre.
Toute valeur de la variable *source* ainsi construite définit un registre. Cette variable est gérée dans les attributs *isource* du pilote et des relations d'archivage.

### 1.3.2 Identification de l'acte dans un registre

Un acte dans un registre est identifiable par son numéro d'ordre séquentiel. La classe *gestion* décrite plus loin permet d'introduire un numéro d'identification attribué en amont dans la phase d'acquisition des données. Ces numéros sont gérés dans l'attribut *iacte*.

### 1.3.3 Rôle des personnes citées

Les personnes citées dans un acte d'état civil ou à sa périphérie ont un rôle dans cet acte géré dans la variable *role*. Ces personnes peuvent

[7] Ces notes sont souvent appelées observations dans les fiches de dépouillement.

TAB. 1.1 - Définition de la variable *evenement*

| Valeur | Signification |
|---|---|
| 0 | indéterminé ou sans objet |
| 10000 | naissance |
| 20000 | mariage |
| 30000 | décès |
| 40000 | divorce |
| 50000 | jugement |
| 60000 | recensement par foyers |
| 70000 | émargement dans une liste |
| 100000 | transcription |
| 110000 | transcription de naissance |
| 120000 | transcription de mariage |
| 130000 | transcription de décès |
| 150000 | reconnaissance d'enfant |
| 200000 | succession |

TAB. 1.2 - Définition de la variable *origine*

| Valeur | Signification |
|---|---|
| 0 | indéterminé ou sans objet |
| 1000 | table décennale |
| 2000 | registre civil |
| 3000 | registre religieux de l'Ancien Régime |
| 4000 | acte notarié |
| 5000 | document juridique |
| 6000 | document fiscal |
| 7000 | document administratif |

aussi avoir un rôle dans un autre acte qui a légalement été établi au vu de l'information lue : date de naissance ou décès, inscription marginale... Cet autre rôle dans un autre acte dit cible est géré dans la variable *cible*.

La structure d'un acte introduit une hiérarchie des rôles des personnes citées avec les niveaux suivants :

1. sujets de l'acte : nouveau-né, mariés ou défunt ;
2. parents, enfants ou conjoints des sujets, c'est-à-dire les personnes ayant un degré de parenté *1* avec un sujet de l'acte ;
3. témoins et autres personnes.

Cette classification demeure utilisable pour d'autres actes.

### Préséance du rôle du sujet

Les règles de transmission des noms de personnes et le mode effectif de choix des témoins font que les patronymes des sujets d'un acte sont aussi des patronymes d'ascendants, descendants et témoins. La mobilité géographique réduite dans les périodes plus lointaines et l'habitat dispersé important observé dans certaines communes rurales font que les lieux désignent parfois des hameaux rattachés au lieu de l'acte. Dès lors que l'orthographe de ces noms propres reste inchangée, le répétiteur du langage $H$ est utilisable.

Si $s$ désigne le rôle d'un sujet et $\mathcal{P}_s$ l'ensemble des rôles $p$ définis par rapport à ce sujet, la propriété

$$\forall p \in \mathcal{P}_s : s < p$$

garantit que les noms du ou des sujets de l'acte sont traités les premiers, pour pouvoir ensuite être répétés. Les conventions adoptées pour coder les rôles doivent être choisies de telle sorte que cette propriété, dite de **préséance du rôle du sujet**, soit toujours vérifiée.

### Codification du rôle

La valeur des rôles pour l'acte source est

$$role = fonction + genre + rang,$$

et pour l'acte cible

$$cible = fonction + genre.$$

TAB. 1.3 - Définition de la variable *fonction*

| Valeur | Signification |
|---|---|
| 1000 | nouveau-né |
| 2000 | marié |
| 3000 | défunt |
| 4000 | divorcé |
| 5000 | parent du nouveau-né, mari ou défunt |
| 6000 | parent de la femme |
| 7000 | conjoint d'un sujet marié antérieurement |
| 8000 | enfant d'un sujet né antérieurement |
| 9000 | témoin |
| 10000 | notaire pour les contrats de mariage |
| 11000 | rédacteur de l'acte : maire, curé, ... |
| 12000 | transcripteur : mise sur fiche, saisie, ... |
| 13000 | autres personnes |
| 20000 | gestion de l'acte |
| 99000 | indéterminé ou inconnu |

La variable *fonction* de la table 1.3 indique la fonction de la personne dans l'acte.

La variable *genre* de la table 1.4 code le sexe, implicite pour certains rôles ou déclaré explicitement pour d'autres : nouveau-né, défunt, témoin.

La variable *rang* $\in [1...99]$ pour certaines valeurs est utilisée pour distinguer les personnes qualifiées par une même valeur de *fonction*, principalement les témoins mais aussi les anciens conjoints et les enfants reconnus.

Les valeurs des variables *role* et *cible* sont gérées respectivement dans les attributs *irole* et *icible*.

TAB. 1.4 - Définition de la variable *genre*

| Valeur | Signification |
|---|---|
| 0 | indéterminé ou sans objet |
| 100 | homme |
| 200 | femme |

Le rôle d'une personne est toujours défini[8] donc positif avec la convention adoptée : la propriété de préséance des rôles des sujets dans les actes est vérifiée.

## 1.4 Identification des variables

D'un point de vue pratiqne, un acte d'état civil est un texte relativement stucturé rédigé selon un protocole adapté aux exigences légales et aux usages. Le contenu d'un acte peut en conséquence être transcrit dans des fiches qui regroupent les variables les plus communes, les autres pouvant toujours être saisies sous forme de texte libre.

Cette transcription dans les fiches traduit un premier niveau de codage réalisé par les experts d'une discipline. Ainsi, la formule « *Le huit Vendémiaire an cinq de la République sont comparus Etienne Poupa tisserand au Bressac en cette commune et Magdeleine Mercier son épouse ...* » nomme deux personnes physiques, précise le métier du mari ainsi qu'un nom de hameau administrativement rattaché au lieu de l'acte.

Les informations associées à ces expressions sont les valeurs de variables définies sur l'ensemble $\mathcal{H}$ et codant ici une date, deux noms propres, un métier et un lieu. On en déduit également que ces deux personnes sont vivantes et mariées à la date d'établissement de l'acte.

Les attributs descriptifs des variables, définis sur $\mathcal{N}$, sont préfixés par la lettre $j$.

### 1.4.1 Classes de variables

Les variables définies sur $\mathcal{H}$ sont réparties dans des classes selon lenr nature : la classe d'appartenance, définie sur l'intervalle [1...NBCLASSE], est codée dans la variable *classe* gérée dans l'attribut *jclasse*. Les valeurs de *classe* situées en dehors de cet intervalle sont utilisées pour gérer d'autres variables.

#### Description des classes

Les NBCLASSE classes sont nommées dans le tableau 1.5.

Ces classes sont des conteneurs d'objets d'une même famille et leur utilisation n'est pas restreinte aux données de l'état civil : la classe STATUT est

[8] Les observations diverses sont affectées par exemple au transcripteur.

TAB. 1.5 - Classes des variables

| Nom | Numéro | Intitulé |
|---|---|---|
| PERSONNE | 1 | identification de la personne |
| PERIODE | 2 | date absolue ou relative |
| LIEU | 3 | localisation géographique |
| STATUT | 4 | métiers, fonctions, titres, diplômes, signature, ... |
| PARENT | 5 | lien de parenté |
| MESURE | 6 | représentation décimale d'une mesure |
| CITATION | 7 | mots quelconques du langage $H$ |
| GESTION | 8 | classification et identification de l'acte |

par exemple disponible pour gérer des listes quelconques et la classe MESURE introduite pour traiter ultérieurementles données patrimoniales.

La classe GESTION regroupe des variables de gestion de plans de classement et de clés d'identification.

### Variables PAGI hors classes

La valeur nulle de la variable *classe* désigne une variable artificielle utilisée exclusivement pour coder le type d'évènement civil cité dans les mentions marginales des actes dans l'environnement PAGI.

Les valeurs supérieures à NBCLASSE désignent des textes libres à archiver alors qu'une valeur négative permet d'ignorer le texte lu.

### Préséance de la classe personne

Une propriété dite de **préséance de la classe personne** est définie afin que la classe d'identification de la personne soit traitée avant les autres classes, comme indiqué au paragraphe 2.1 : la classe PERSONNE est en conséquence de rang 1.

Cette propriété est introduite pour pouvoir calculer les rôles des personnes citées, qui peuvent dépendre du sexe du sujet de l'acte, avant de traiter les variables des autres classes.

### 1.4.2 Types de variables

Les variables historiques sont désignées dans les classes par un code qui précise la nature de l'information représentée: lieu d'origine ou de résidence, date de naissance ou décès, nom légal ou d'usage, ...

Pour l'état civil, ce code, géré dans l'attribut *jtype*, prend en considération simultanément

- l'évènement lié à une mention marginale ou une date,
- le rang d'une mention marginale ou d'un élément de liste,
- la nature d'une variable des classes PERSONNE, PERIODE ou LIEU,
- la forme lue dans la classe PERIODE,
- le type de subdivision dans la classe LIEU,
- le système d'unités,
- le sujet auquel se rapporte un lien de parenté,
- le plan de classement choisi.

Sa valeur est la somme des variables intermédiaires

$$type = evenement + nature + forme + cause + unite + sequence.$$

Toutes ces variables reçoivent la valeur nulle par défaut.

*evenement* est définie plus haut dans la table 1.1 ;

*nature* est définie en fonction des classes dans la table 1.6 ;

*forme* est définie pour la classe PERSONNE dans la table 1.7 ;

*cause* est définie pour la classe LIEU dans la table 1.8 ;

*unite* est définie pour la classe PERIODE dans la table 1.9 : si plusieurs unités sont effectivement présentes ($forme = 300$), c'est l'unité la plus fine qui est retenue ;

*sequence* contient le numéro séquentiel d'une mention marginale ou d'un élément de liste.

TAB. 1.6 - Définition de la variable *nature*

| **Classe** | Valeur | **Signification** |
|---|---|---|
| PERSONNE | 1000 | nom légal |
| | 2000 | indication époux ou épouse |
| | 3000 | indication veuf ou veuve |
| | 4000 | nom de naissance indiqué explicitemment |
| | 5000 | surnom |
| PERIODE | 1000 | date exacte |
| | 2000 | âge |
| | 3000 | délai |
| | 4000 | délai indéterminé |
| LIEU | 1000 | commune |
| | 2000 | paroisse |
| | 3000 | hameau |
| | 4000 | groupe de hameaux |
| | 5000 | quartier |
| | 6000 | écart |
| | 9000 | adresse |

TAB. 1.7 - Définition de la variable *forme*

| **Valeur** | **Signification** |
|---|---|
| 100 | nombre mesurant des années |
| 200 | nombre avec code unité<br>*2m* pour deux mois |
| 300 | suite de nombres avec codes unités<br>*2m 3j* pour deux mois et trois jours |
| 400 | indication majeur |
| 500 | indication mineur |
| 600 | indication vivant |
| 700 | indication mort |

TAB. 1.8 - Définition de la variable *cause*

| Valeur | Signification |
|---|---|
| 100 | lieu d'origine |
| 200 | lieu de résidence |
| 300 | lieu de travail |
| 400 | localisation d'une propriété |

TAB. 1.9 - Définition de la variable *unite*

| Valeur | Signification |
|---|---|
| 10 | calendrier grégorien, |
| 20 | calendrier républicain, |
| 30 | calendrier julien, |
| 50 | années, ( *1a* pour un an), |
| 60 | mois, ( *8m* pour huit mois), |
| 70 | semaine, ( *3s* pour trois semaines), |
| 80 | jour, ( *5j* pour cinq jours), |
| 90 | heure, ( *1h* pour une heure). |

## 1.5 Espace de traitement des variables historiques

Avec les conventions précédentes, tout élément $h_{ij} \in \mathcal{H}$ est identifié au moyen de la clé

$$(isource, iacte, irole, jclasse, jtype).$$

Dans l'environnement d'acquisition des données de l'état civil PAGI, cet élément est construit par assemblage de composants.

### 1.5.1 Construction des variables

Les composants PAGI sont des champs définis empiriquement par rapport à une logique de saisie des données de l'état-civil traduite dans un format.

#### Numéro PAGI

Les champs sont séparés par une marque FIN_VAR de fin de variable et la fin de l'acte est repérée par la marque FIN_ACTE : l'absence de valeur pour une variable est traduite par deux marques consécutives.

Un champ peut ainsi être localisé par son rang séquentiel dans l'acte : c'est le numéro PAGI géré dans l'attribut *jpagi*.

#### Format PAGI

Les composants PAGI sont définis pour une suite fixe de rôles de personnes normalement citées dans l'acte.

Le format PAGI prend en compte un certain nombre d'usages : une date calendaire est décrite par le jour, le mois et l'année, avec des variantes d'assemblage de ces trois composants ; un lieu d'établissement d'un acte d'état civil en France est une commune d'un département, aujourd'hui repérée par un code INSEE unique et une orthographe officielle.

La syntaxe relativement libre conduit à adopter moult conventions pour pouvoir indiquer la uature exacte de l'information ou préciser qu'une incertitude demeure. Ainsi, des toponymes sont cités sans qu'on sache s'il s'agit d'un hameau ou d'une commune, ou à quelle commune est rattaché le hameau désigné. Pour la période antérieure à la Révolution, le nom de la paroisse ou du Saint-Patron remplace celui de la commune.

Ces usages et les degrés de liberté associés sont pris en compte dans les classes au moyen d'un code géré dans l'attribut *jfpagi* : Les conventions actuelles sont décrites en annexe A.

### 1.5.2 Description des variables

Les variables sont ici regroupées en catégories en fonction de leur signification pratique. La liste ci-dessous reprend le modèle de données utilisé à l'INRA pour l'état civil.[9]

#### 1. Identifiant civil

Toute personne physique est identifiée dans la vie civile par son nom et ses prénoms, qui définissent une première variable d'état de la classe PERSONNE dont la valeur est un identifiant civil, affecté dans l'acte initial de naissance, normalement invariant dans le temps et l'espace.

Les valeurs de cette variable pour une même personne dans les actes anciens sont souvent transcrites sous des formes graphiques diversifiées : variantes orthographiques, listes de prénoms partielles ou permutées, introduction de surnoms, ... Il est en conséquence nécessaire de calculer des équivalences pour pouvoir comparer deux identifiants.

Notons également que cet identifiant est souvent incomplet dans les documents les plus anciens : la valeur transcrite est alors indéterminée et il n'est pas possible de calculer des équivalences.

#### 2. Sexe

Le sexe est la seconde variable d'état de la classe PERSONNE dont la valeur absolue est déterminée à la naissance et invariante dans le temps. En revanche, l'information initiale peut, exceptionnellement, être erronée.

#### 3. État matrimonial

L'état matrimonial est une variable d'état de la classe PERSONNE qui change de valeur à l'issue d'un mariage, du décès du conjoint ou d'un divorce.

#### 4. Légitimité

Des personnes physiques différentes peuvent être désignées par les mêmes noms et prénoms d'où la nécessité de prendre en compte la filiation lorsqu'elle est établie pour lever d'éventuelles ambiguïtés. La filiation est par-

[9] **Jean-Claude Poupa**, *Les procédures et les moyens informatiques pour l'enquête sur les 3000 familles "TRA"*, INRA, Economie et Sociologie, Rennes, 1993.

tielle pour l'enfant naturel ou parfois totalement inconnue au moment de la naissance. Elle peut être établie plus tard lors d'une reconnaissance d'enfant.

La légitimité de l'enfant est une variable d'état de la classe PERSONNE dont la valeur figure souvent pour les sujets, parents, enfants et conjoints : dans le contexte PAGI, elle n'est retenue que pour le nouveau-né.

### 5. Date

Les évènements surviennent en un jour d'un temps calendaire identifié au moyen d'une date dans une variable de la classe PERIODE. L'établissement de l'acte associé à cet évènement s'effectue simultanément pour les mariages et divorces et postérieurement pour les naissances et décès.

Le délai entre l'évènement et l'établissement de l'acte, noté dans une composant PAGI spécifique, est utilisée pour calculer la date précise de l'évènement si ce délai est connu. Dans le cas contraire, la date de l'évènement n'est pas totalement déterminée.

La propriété de préséance du rôle du sujet énoncée au paragraphe 1.3.3 garantit que la date de l'évènement générateur de l'acte est traitée avant les autres dates. Elle permet de vérifier que les dates citées dans le corps de l'acte sont antérieures à cet évènement alors que celles citées en marge sont postérieures, ou égales pour le décès du nouveau-né le jour de sa naissance.

### 6. Âge

L'âge d'une personue, qu'il soit exprimé par un nombre d'années ou déduit d'une indication, mesure un intervalle de temps entre sa naissance et la date d'établissement de l'acte dans laquelle elle est citée. Il est géré dans une variable de la classe PERIODE sous différentes formes et au moyen de plusieurs systèmes d'unités. Plusieurs valeurs plus ou moins précises peuvent figurer dans un acte pour une même personne : la valeur du type d'information permet alors de différencier ces valeurs.

### 7. Lieu

Les lieux d'établissement des actes sont identifiés par des toponymes, qui désignent des parties d'un territoire circonscrites par des limites administratives susceptibles d'évoluer dans le temps. Un même toponyme peut désigner plusieurs lieux. Afin de lever ces ambiguïtés, il est d'usage de citer des entités administratives auxquelles sont rattachées ces lieux, par exemple l'arrondissement et le département pour les communes françaises. Pour la

période récente, les noms officiels des communes et les codes des départements sont suffisants pour identifier tous les lieux d'établissement des actes.

Le lieu de survenance de l'évènement est souvent désigné en zone rurale par un autre toponyme local qui correspond à une partition plus fine du territoire en hameaux ou écarts pour localiser les lieux habités. Cette relation d'inclusiou est traduite en plaçant l'élément inclus entre parenthèses : *Coullons(La Sasserie)* désigne un hameau d'une commune.

Les lieux cités daus les actes sont associés à une personne et un évènement au sens large : état civil, résidence, travail, etc. Le département et la commune sont nécessairement présents.

La propriété de préséance du rôle du sujet garantit, comme pour les dates, que le lieu d'établissement de l'acte est traité avant les autres lieux.

### 8. Activités

Les actes contiennent la plupart du temps pour les personnes majeures des informations relatives à leurs situations dans la vie sociale au moment de la rédaction de l'acte, généralement un métier, parfois une fonction ou un titre. Ces renseignemeuts sont transcrites sous la forme d'une expression ou d'une liste d'expressions : l'information élémentaire, gérée dans une variable de la classe STATUT, est relative à une activité d'une personne citée.

### 9. Signature

Certaines personnes citées dans les actes ont la possibilité de signer : l'existence de cette signature est codée dans une variable spécifique de la classe STATUT.

### 10. Liens avec les sujets

Des informations relatives aux liens de parentés ou autres des personnes citées avec les sujets de l'acte figurent parfois en clair, notamment pour les témoins : les termes explicitant ces liens sont conservés dans une liste.

Un lien est géré dans une variable de la classe PARENT et défini pour une personne par rapport à un sujet. Il peut exister plusieurs liens pour une même personne par rapport aux sujets.

11. **Autres valeurs**

Une variable spécifique de la classe CITATION permet de conserver les mots du langage $H$ en vue de traitements ultérieurs ou à des fins documentaires. Les valeurs sont transcrites après application des règles de normalisation définie au paragraphe 1.1.4.

### 1.5.3 Suivi des acquisitions

Un acte est établi à une date et en un lieu pour des sujets — nouveau-né, mari et femme, défunt — désignés par un nom. Ces valeurs sont également utilisées pour classer les actes de l'état civil dans les tables décennales.

#### Tables décennales réelles et calculées

Les actes disponibles sont situés dans l'espace et le temps et classés en fonction des patronymes à partir des données lues. Celà revient à calculer les tables décennales des actes dépouillés.

Les tables décennales réelles sont par ailleurs traitées comme un ensemble d'actes pour lesquels les seules variables observées sont la date, le lieu et les noms des sujets.

La comparaison des données calculées et réelles gérées dans des relations permet de réaliser des bilans et de définir des tests de contrôle d'exhaustivité comme de non répétition.

#### Classement des actes

La classe GESTION a été introduite pour pouvoir classer les actes selon tout autre critère, déjà défini dans une classe ou déclaré comme plan de classement dans le pilote. Il est ainsi possible d'effectuer des classement tant en fonction des lieux que des patronymes.

# Chapitre 2

# Représentation formelle des valeurs historiques

Les valeurs atomiques d'une variable historique sont gérées dans une ou plusieurs relations.

Les systèmes de gestion de bases de données relationnelles classiques imposent, eu égard au modèle théorique, des restrictions fortes. Si les technologies objets sont aujourd'hui en mesure de réduire ces contraintes, la mise en œuvre d'une solution avec les logiciels actuels demeure complexe. En tout état de cause, il est nécessaire de spécifier dans le détail le modèle de calcul afin de pouvoir, à défaut de disposer du logiciel idoine, le « programmer » en utilisant au mieux les technologies disponibles.

Une première restriction forte liée aux logiciels relationnels du commerce concerne le choix des domaines : un ensemble de mots, qu'il soit énuméré ou généré par une grammaire, est vu comme toutes les valeurs textuelles comme un tableau de caractères.[1] L'ensemble $\mathcal{H}$ ne peut donc pas être déclaré directement comme domaine. Dans un tel environnement, les calculs, basés sur la comparaison des éléments de même rang de deux opérandes, sont inefficaces pour de grands ensembles. Pour réduire cette complexité, les composants syntaxiques d'une même valeur sont séparés et gérés dans une ou plusieurs relations afin d'aboutir à une représentation logique et physique sur laquelle les machines relationnelles actuelles sont effectivement plus efficaces.

Une seconde restriction est imposée par cette impossibilité de construire des domaines : il faut se donner une limite supérieure pour la longueur des

[1] Dans la théorie des langages, avec l'opérateur de concaténation des caractères d'un alphabet $\mathcal{A}$, ce domaine est le monoïde libre noté $\mathcal{A}^*$.

chaînes de caractères, alors que cette limite n'existe pas dans les définitions théoriques. Ce problème est résolu par la mise en place d'un mécanisme de gestion des exceptions afin de conserver l'intégralité de l'information en cas de dépassement d'une borne supérieure.

C'est dans ce contexte sont définies des relations théoriques dites **canoniques**, utilisables pour construire d'autres relations. Toujours pour des raisons d'efficacité, ces relations contiennent au plus un attribut textuel, les autres attributs étant numériques : une relation de degré $k$ permet de gérer une partie du produit cartésien $\mathcal{N}^{k-1} \times \mathcal{H}$ ou de $\mathcal{N}^k$.

Les noms lus dans les actes appartiennent généralement à des ensembles finis gérés dans des **dictionnaires**, construits au moyen des mêmes instruments afin de pouvoir vérifier dans une étape ultérieure l'existence des noms effectivement transcrits dans les actes.

Certaines variables dans des actes signalent l'existence d'évènements — naissances, mariages, décès, divorces — qui ont normalement donné lieu à l'établissement d'actes. Ces **références** vers d'autres actes sont codées sous forme de clés numériques dans une relation spécifique.

## 2.1 Assemblage des composants PAGI

Un composant PAGI repéré par un numéro en séquence possède un format et est lié à une variable historique d'un type donné dans une classe. Il contient une valeur relative au rôle d'une personne dans un acte. Cette description élémentaire est représentée par un sexuplet du pilote $\mathcal{P}$ géré dans la relation

$$\mathbf{P}(\mathbf{isource}, irole, \mathbf{jclasse}, jtype, \mathbf{jpagi}, jfpagi).$$

La clé primaire permet de déclarer un même composant PAGI dans plusieurs classes et garantit l'unicité de l'instruction associée. Ainsi, les codes numériques des lieux d'établissement des actes gérés dans la classe LIEU sont aussi utilisables dans la classe GESTION pour produire des plans de classement des actes comme indiqué au paragraphe 3.9.

Le pilote associé au registre $\sigma$ est l'ensemble $\mathcal{S}_t$ des sexuplets qui vérifient à l'instant $t$ le prédicat $(isource = \sigma)$. Il contient les instructions transmises au compilateur pour assembler les composants.

### 2.1.1 Ordonnancement des directives

L'ensemble ordonné

$$\mathbf{S}_{1t}(\mathbf{isource}; \mathbf{jclasse}; irole; jtype, \mathbf{jpagi}; jfpagi)$$

fournit la suite des instructions de traitement des variables de l'acte.

#### Préséance de la classe PERSONNE

La première valeur non nulle de *jclasse* désigne, avec la définition de la table 1.5, la classe PERSONNE. La valeur exacte de la variable *role* est calculée et conservée pour identifier les valeurs dans les autres classes. Elle est également utilisée pour calculer la variable *cible* lorsque des évènements civils sont cités en marge de l'acte, pour le sujet et son conjoint.

#### Préséance du rôle du sujet

La valeur contenue dans *irole* est strictement positive. Les variables qui décrivent les sujets sont traitées en premier dans leurs classes respectives. Les patronymes et les lieux peuvent ainsi être répétés. La date de l'acte est utilisée pour contrôler la chronologie des évènements pour les autres rôles et permet aussi de calculer des dates d'évènements à partir des âges.

### 2.1.2 Directives liées

Les date et lieu d'un mariage ou divorce sont identiques pour les deux sujets de l'acte : il suffit de connaître l'une de ces directives pour générer automatiquement la seconde. On évite ainsi de saisir explicitement deux directives dont il faudrait alors vérifier la compatibilité.

### 2.1.3 Calcul des rôles

Le codage des rôles, décrit au paragraphe 1.3.3, utilise les variables intermédiaires *fonction*, *genre* et *rang* qui ont une valeur définie dans un acte. Le rôle de la personne dans l'acte courant, dit source ou d'origine, est codé dans la variable *role*. La variable *cible* code un rôle de sujet dans un autre acte à rechercher, dit référencé ou d'arrivée. Par construction, la variable *fonction* est toujours positive. En conséquence, les valeurs calculées des variables *role* et *cible* sont toujours positives.

La valeur de *role* peut être initialisée par lecture du pilote puis calculée définitivement après lecture du contenu de l'acte. Pour les mentions marginales et les dates, ce rôle est déduit du type d'évènement codé dans un composant PAGI spécifique.

### Rôle dans l'acte source

Les conventions PAGI font que les valeurs des variables *fonction* et *rang* sont connues a priori pour tout composant PAGI. La variable *genre* est prédéterminée pour les pères et mères ou indéterminée. En conséquence, l'attribut *irole* du pilote pour la variable $j$ reçoit la valeur

$$s_j = fonction + genre + rang,$$

avec $genre = 0$ si le sexe est a priori inconnu.

Le codage choisi permet de retouver les valeurs des variables intermédiaires. Si $\gamma$ représente la valeur de la variable *genre* codée dans $s_j$, le sexe de la personne sera déterminé en fonction des valeurs lues dans l'acte seulement si $(\gamma = 0)$, ce que traduit l'expression conditionnelle

$$\text{si } (\gamma = 0) \text{ alors } s_{ij} = s_j + genre \text{ sinon } s_{ij} = s_j.$$

Le sexe effectif, codé dans la variable *genre*, dépend de la valeur lue de *fonction* : un conjoint est de sexe opposé à celui du sujet.

Dans un acte de mariage ou divorce, les sujets sont mariés ou divorcés, et non conjoints avec les conventions retenues. Le sexe est généralement codé dans l'attribut *irole* du pilote : il n'est pas inutile de vérifier que ces sujets sont effectivement de sexes opposés.

### Rôle dans l'acte référencé

La variable *cible* gère un rôle de sujet dans un autre acte, traduit par la valeur $\varphi \in \{1000, 2000, 3000, 4000\}$ de la variable *fonction*, déduite de la valeur $\beta$ de l'évènement.

La variable *rang* est sans objet puisque *cible* code un rôle de sujet, unique par définition. La valeur de la variable *genre* demeure identique à celle déduite de l'acte source. Le rôle dans l'acte référencé est déduit en totalité du contenu de l'acte.

## 2.1.4 Calcul des types

La valeur du type d'information pour la variable $j$ est la somme

$$t_j = evenement + nature + forme + cause + unite + sequence$$

La variable *evenement* est, pour une mention marginale ou une date, lue dans un composant PAGI. Elle est en revanche déterminée en fonction de la valeur de la variable *nature* pour les lieux et l'âge.

Les variables *nature*, *cause* et *sequence* sont constantes pour un registre et déterminées a priori : la valeur du type $t_j$ dans l'acte source, gérée dans l'attribut *jtype* du pilote pour la variable $j$ est la somme

$$t_j = nature + cause + sequence.$$

Les variables $forme$ et *unite* sont déduites du contenu de l'acte.

Finalement, la valeur du type d'information pour un fait élémentaire décrit par la variable $j$ dans l'acte $i$ est la somme

$$t_{ij} = t_j + evenement + forme + unite.$$

## 2.2 Construction des relations d'archivage

L'absence de « moteur relationnel » efficace pour évaluer des expressions relationnelles sur de grands ensembles de données non numériques conduit à définir plusieurs relations, les unes utilisables comme opérandes dans des séquences automatisées de calcul, les autres destinées à restituer l'intégralité de l'information en cas de besoin.

Un moyen pour réduire la complexité est d'isoler les noms de la langue et de traiter les occurrences rares avec les exceptions. Celà revient à restreindre aux ensembles $\mathcal{N}$ et $\mathcal{G}$ les domaines des attributs des relations opérandes, en d'autres termes à travailler sur les ensembles de nombres et les dictionnaires.

### 2.2.1 Extraction des noms et troncatures

La démarche naturelle de reconnaissance d'une phrase passe par une analyse syntaxique qui isole les noms : la séquence

« *Poupa(dit Cadet), Célestin Isidore* »

déjà citée est éclatée en quatre noms :

« *Poupa* » est le patronyme,
« *dit* » signale que le mot suivant, « *Cadet* », est un surnom,
« *Célestin* » et « *Isidore* » sont deux prénoms.

Sur un plan plus formel, ces noms sont les mots dérivés de la variable syntaxique $< hnom >$ : ils définissent l'ensemble $\mathcal{G}$ du paragraphe 1.1.4.

Une valeur $\mu$ de $\mathcal{H}$ est dite

- **simple** si $\mu \in \mathcal{G}$,
- **complexe** si $\mu \notin \mathcal{G}$.

La longueur maximale des noms étant fixée, il reste à définir une méthode de segmentation pour les valeurs qui dépassent cette longueur.

### Définition des opérateurs de segmentation

La segmentation décrite dans l'exemple qui précède peut se formaliser au moyen de deux opérateurs unaires $\theta_1$ et $\theta_2$ définis sur $\mathcal{H}$, le premier rendant le nom situé à gauche d'un caractère dit de segmentation, le second le reste du texte. Les caractères de segmentation, dits *points de rupture*, sont les ponctuations et l'espace.

L'ensemble des points de rupture est formellement une relation unaire avec un attribut défini sur l'alphabet du langage $H$.

Soit $\varphi \in \mathcal{H}$ et $\epsilon$ l'élément neutre de $\mathcal{H}$.
Le premier nom est extrait par la formule

$$\varphi_1 = \theta_1(\varphi)$$

et les noms de rang $k > 1$ sont évalués par la règle

$$\text{si } (\theta_2(\varphi_k) \neq \epsilon) \text{ alors } \varphi_{k+1} = \theta_1(\theta_2(\varphi_k)).$$

A l'issue de ces opérations, la valeur initiale $\varphi \in \mathcal{H}$ est décomposée en éléments $\varphi_k \in \mathcal{H}$ qui représentent les noms extraits du texte géré dans $\varphi$.

### Choix des points de rupture

La restitution des noms réels gérés dans les variables historiques nécessite de conserver les espaces dans un patronyme, éventuellement l'apostrophe. Il faut également pouvoir paramétrer des stratégies de rupture, afin par exemple de prendre en compte les prénoms composés en distinguant le tiret de l'espace dans une liste de prénoms.

Cette fonctionnalité est assurée en associant aux variables historiques l'ensemble des caractères à retirer de l'ensemble des points de rupture. Ce ensemble, appelé *masque*, est vide par défaut.

### Troncatures

Une longueur maximale $L_j$ est définie pour chaque variable $j$ : si la longueur effective du mot dépasse cette valeur, il est nécessaire d'utiliser les opérateurs précédents.

Afin d'éviter un éclatement en de multiples noms le point de rupture choisi est celui qui conserve les mots les plus longs générés par le langage

$H$. Pour ce faire, l'opération est réalisée en isolant le composant syntaxique $< groupe >$ et en positionnant la variable booléenne *inclusion* au moyen de la règle

Si $< groupe >$ dérive de $< inclusion >$ alors *inclusion* = `vrai`
sinon *inclusion* = `faux`.

Le mot dérivé de $< groupe >$ est conservé tel quel si sa longueur effective le permet, segmenté en utilisant comme point de rupture le composant syntaxique $< hsepa >$ — espace, tiret, apostrophe ou barre oblique — qui précède le caractère de rang $L_j$ dans le cas contraire. La règle est appliquée comme pour toute segmentation de façon récursive tant que $(\theta_2(\varphi_k) \neq \epsilon)$.

### Indéterminations

Si le point de rupture recherché pour une valeur affectée dans la variable $j$ n'existe pas, les opérateurs $\theta_1$ et $\theta_2$ ne sont plus définis sur $\mathcal{H}$. Cette situation, qui génère une erreur, se produit pour toute occurrence d'un nom en usage dans une langue nationale dont la longueur, mesurée en nombre de caractères, dépasse la valeur maximale $L_j$. Le fait de masquer certains caractères, par exemple le trait d'union des prénoms composés, risque d'introduire de nouvelles indéterminations.

Sur un plan pratique, les constantes $L_j$ sont choisies de telle sorte que les noms les plus longs construits avec l'alphabet de la langue puissent être gérés. Dans la mesure où les performances dépendent des valeurs choisies, il peut être judicieux d'exclure certains noms très rares.[2]

### Codage des opérations

Afin de pouvoir reconstruire ultérieurement le texte initial, la variable *rupture* gère la segmentation. Elle est initialisée comme indiqué dans le tableau 2.1 et sa valeur est ensuite multipliée par un facteur 100 si la variable *inclusion* vaut `vrai`, conservée sinon.

[2]Si la variable $j$ gère les prénoms en usage, le choix de la valeur $L_j = 16$ ne permet pas de gérer quelques prénoms composés très rares effectivement rencontrés comme « Marie-Alexandrine » et « Marie-Guillemette » pour lesquels on dénombre 17 caractères. Le fait d'augmenter la constante $L_j$ se traduirait très vraisemblablement sur la plupart des machines par une dégradation globale des performances de l'ordre de 25%, pénalisante puisque tous les prénoms des personnes citées sont gérées dans une même relation.

TAB. 2.1 - Initialisation de la variable *rupture*

| composant | caractère | code |
|---|---|---|
| < *delimiteur* > | , | 1 |
| | : | 2 |
| | ; | 3 |
| < *separateur* > | - | 4 |
| | ' | 5 |
| | / | 6 |
| < *espace* > | | 7 |

### 2.2.2 Gestion des exceptions

Les exceptions sont levées dans plusieurs contextes :

- la longueur du mot dépasse la valeur maximale de la classe,
- une note placée entre parenthèses apporte une précision,
- une information est transcrite sous une forme syntaxique rare,
- un cas de figure non répertorié se présente.

Le texte initial est découpé en mots de l'ensemble $\mathcal{H}$, appelés **segments**, au moyen des opérateurs d'extraction des noms décrits au paragraphe 2.2.1.

#### Déclenchement des exceptions

Le composant syntaxique < *inclusion* > ne doit être utilisé que pour saisir des informations rares ou imprévisibles : dérogations civiles ou religieuses, ambiguïtés signalées lors de la transcription, surnom d'une personne... Si ces informations sont relativement fréquentes, il est opportun de créer une variable spécifique.

Les longueurs maximales des noms sont fixées dans les classes au vu de statistiques sur les données réelles. Les valeurs doivent être suffisamment grandes pour gérer la quasi-totalité des noms. Ce sont des constantes choisies en fonction du contexte applicatif :

- une valeur trop grande se traduit par une cousommation importante de ressources et un effondrement des performances ;

- une valeur trop faible augmente les performances mais réalise les calculs dans un univers réduit, avec le risque d'établir de fausses équivalences.

#### Archivage du texte initial

Le texte segmenté est géré dans une relation annexe au moyen de trois attributs spécifiques :

*k* code le numéro en séquence du segment ;
*rupture* gère la variable *rupture* ;
*hsegment* reçoit le segment.

La clé $(isource, iacte, irole, jtype, jclasse)$ identifie la valeur à laquelle est liée l'exception. Le texte initial est archivé dans la relation

$$\mathbf{E}(\mathbf{isource}, \mathbf{iacte}, \mathbf{irole}, \mathbf{jtype}, \mathbf{jclasse}, \mathbf{k}, rupture, hsegment).$$

La relation d'ordre notée

$$\mathbf{E_1}(\mathbf{isource}; \mathbf{iacte}; \mathbf{irole}; \mathbf{jtype}; \mathbf{jclasse}; \mathbf{k}; rupture, hsegment)$$

génère l'ensemble ordonné $\mathcal{E}_{1t}$ des exceptions à l'instant $t$.

#### Codage de l'exception dans les relations d'archivage

L'existence d'informations complémentaires est codée dans l'attribut *suite* des relations d'archivage qui reçoit alors le nombre total de segments. En l'absence d'exception, cet attribut reçoit une valeur nulle.

### 2.2.3 Archivage des valeurs simples

La nécessité d'optimiser les calculs conduit à décomposer les valeurs complexes en valeurs simples définies sur $\mathcal{G}$ ou $\mathcal{N}$ et gérées dans des relations utilisables comme opérandes dans des expressions relationnelles à évaluer. Si une exception est levée, la valeur est conservée telle quelle sous sa forme complexe pour pouvoir être consultée à la demande.

### Décomposition syntaxique

L'analyse syntaxique permet d'isoler des nombres et des noms qui se répartissent dans différentes catégories en fonction des traitements à réaliser.

- Des mots en usage dans la langue définissent des ensembles de quelques éléments auxquels sont immédiatement associés des codes numériques : sexe, état matrimonial, légitimité, signature.
- Des noms simples ou composés ont une valeur sémantique bien établie : patronyme, prénom, toponyme, métier ou fonction, lien de parenté. Ils appartiennent à des ensembles finis mais dont les cardinalités se mesure en milliers, voire millions.
- Les dates sont des suites de trois nombres codant le jour, le mois et l'année dans le calendrier grégorien.

### Représentation relationnelle

La méthode de décomposition est illustrée à partir du format PAGI d'acquisition des noms et prénoms dans lequel

> la virgule sépare le patronyme de la liste des prénoms,
> l'espace sépare les prénoms.

La valeur complexe « *Poupa,Etienne Pierre* » relative au nouveau-né de l'acte numéro $\alpha$ du registre $\sigma$ est décomposée en valeurs simples.

- Le patronyme est représenté par le quintuplet

$$(\sigma,\ \alpha,\ 1000,\ 0,\ \ll Poupa \gg)$$

de la relation

**Nom**(**isource**, **iacte**, **irole**, **jtype**, *suite*, *g*),

l'attribut *suite* codant les exceptions relatives à cette valeur complexe.

- Les prénoms sont gérés dans les sexuplets

$$(\sigma,\ \alpha,\ 1000,\ 0,\ 1,\ \ll Etienne \gg)$$
$$(\sigma,\ \alpha,\ 1000,\ 0,\ 2,\ \ll Pierre \gg)$$

de la relation

**Prenom**(**isource**, **iacte**, **irole**, **jtype**, **rang**, *g*),

l'attribut *rang* contenant le rang du prénom dans la liste.

Les relations définies dans les classes sont décrites au chapitre 3.

### 2.2.4 Archivage des valeurs complexes

L'éclatement des valeurs complexes peut aussi être différé compte-tenu des priorités et des objectifs. L'information est alors conservée sous sa forme initiale dans la classe CITATION. Cette classe peut aussi gérer les notes et remarques complémentaires relatives à l'acte ou à une variable pour laquelle l'expression parenthèsée n'est pas autorisée ou code explicitement une inclusion.

Les valeurs complexes de la classe CITATION sont gérées selon un mode analogue à celui des exceptions par segmentation du texte, comme décrit au paragraphe 2.2.1, dans la relation

$$\mathbf{C}(\mathbf{isource}, \mathbf{iacte}, \mathbf{irole}, \mathbf{jtype}, \mathbf{k}, \mathit{suite}, \mathit{hsegment}).$$

### 2.2.5 Regénération des actes

Les ensembles de faits élémentaires d'un registre $S$ identifié par la valeur $\sigma$ de la variable *source* sont calculés par projection de l'attribut *isource* après restriction par le prédicat $(isource = \sigma)$.

Dans un registre , les faits propres à un acte $A$ identifié par la valeur $\alpha$ de la variable *acte* sont extraits par projection de l'attribut *iacte* après restriction par le prédicat $(iacte = \alpha)$.

Un acte peut être regénéré par assemblage des variables dans un format choisi. Le regroupement des segments s'effectue ensuite en fonction des valeurs de l'attribut *suite*.

## 2.3 Lexiques et dictionnaires

Le langage $H$ assemble de noms. Pour les données de l'état civil, ces noms sont des patronymes, prénoms, toponymes, métiers, fonctions, nommages de liens de parenté. Ils peuvent être regroupés dans des listes ou dans des structures hiérarchiques, comme « *nom, prénom1, prénom2* » ou « *commune(hameau)* » .

Les ensembles énumérés des valeurs lues dans les actes pour une variable donnée sont appelés **lexiques**.

Les noms en usage sur une période définissent des ensembles finis gérés dans des **dictionnaires**.

### 2.3.1 Représentation relationnelle

Les lexiques regroupent les valeurs effectives alors que les dictionnaires gèrent des ensembles de noms antérieurement reconnus et validés.

#### Lexiques

Les lexiques sont des formes intermédiaires de thésaurisation des valeurs historiques extraites des relations et gérées dans la relation

$$\mathbf{L}(\mathbf{isource}, \mathbf{iacte}, \mathbf{irole}, \mathbf{jtype}, h).$$

#### Dictionnaires

Les dictionnaires gèrent des ensembles énumérés de noms valides. Ces ensembles peuvent être définis officiellement comme le dictionnaire actuel des communes françaises. Ils sont cependant la plupart du temps construits par union de sous-ensembles d'origines diverses sans qu'il soit possible de reconstituer l'ensemble fini des noms effectivement en usage: c'est le cas pour les patronymes et les toponymes de façon globale.

La structure d'un dictionnaire est illustrée sur l'exemple des communes françaises à partir d'un dictionnaire hypothétique qui serait géré dans la relation

$$\mathbf{Communes}(nom, \mathbf{departement}, \mathbf{commune}, \mathbf{graphie}, postal).$$

Soit un extrait de ce tableau représenté ci-dessous sous la forme tabulaire.

| | | | | |
|---|---|---|---|---|
| *Montfort* | 4 | 127 | 1 | 600 |
| *Montfort* | 35 | 188 | 1 | 160 |
| *Montfort-sur-Meu* | 35 | 188 | 2 | 160 |
| *Montfort-La-Cane* | 35 | 188 | 3 | 160 |
| *La-Nouaye* | 35 | 203 | 1 | 137 |
| *Talensac* | 35 | 331 | 1 | 160 |

L'entrée du dictionnaire est le toponyme géré dans l'attribut *nom* défini sur l'ensemble $\mathcal{G}$. Plusieurs entrées peuvent exister pour des toponymes identiques, ici les « *Montfort* » des Alpes-de-Haute-Provence et d'Ille-et-Vilaine.

Les attributs numériques *departement*, *commune* et *graphie* codent respectivement le département, la commune et l'une des transcriptions graphiques reconnues sur la période de définition du dictionnaire. Ils définissent une clé primaire.

Le couple (*commune*, *departement*) définit une relation d'équivalence : les trois graphies répertoriés pour la commune de « *Montfort* » en Ille-et-Vilaine désignent un même lieu d'établissement des actes identifié par le couple (35, 188). On peut convenir de choisir la valeur 1 pour coder dans l'attribut *graphie* le représentant de la classe d'équivalence, généralement le nom officiel.

L'attribut *postal* contient le code postal de la commune et définit une autre relation d'équivalence, qui regroupe en fait des communes voisines, non suffisante pour retrouver les lieux d'établissement des actes. D'autres attributs peuvent être ajoutés pour prendre en compte d'autres regroupements.

Sur un plan formel, un dictionnaire est géré dans une relation

$$\mathbf{D}(g, \mathbf{n_1}, ..., \mathbf{n_k}, n_{k+1}..., n_p, g),$$

dans laquelle les attributs $n_1, ..., n_p$ définis sur le domaine $\mathcal{N}$ définissent des nomenclatures de classement des toponymes et l'attribut $g$ défini sur $\mathcal{G}$ contient le toponyme. La clé primaire $(n_1, ..., n_k)$ est formée d'une partie des attributs numériques de classement et doit nécessairement être définie.

### 2.3.2 Construction des dictionnaires

Les dictionnaires sont construits par lecture directe d'un registre dans lequel l'acte contient une seule variable textuelle définie sur $\mathcal{H}$ et optionnellement des variables numériques de classification. Ces registres peuvent être d'origines exogènes, par exemple le dictionnaire officiel des communes, ou issus d'énumérations des valeurs effectivement lues dans les actes.

Comme indiqué plus haut, les entrées d'un dictionnaire sont des mots de l'ensemble $\mathcal{G}$. En conséquence, les listes sont segmentées au niveau des délimiteurs et les inclusions sont extraites. Les listes dans les inclusions sont également segmentées. Ce traitement rend la liste des mots extraits de la valeur initiale définie sur $\mathcal{H}$. La production d'un dictionnaire nécessite de définir des directives de pilotage transmises au compilateur du langage $H$. Les directives définies plus haut sont utilisables mais il est possible dans ce contexte de définir un pilote simplifié.

#### Pilote

La variable dont les valeurs définissent un dictionnaire appartiennent à une classe qui peut être identifiée par une simple valeur d'option transmise

en paramètre. Les variables *role* et *type* sont sans objet puisque la valeur est unique. Le nombre restreint de dictionnaires utiles conduit à nommer ces dictionnaires plutôt que de les identifier par un code numérique, ce qui rend inutile l'attribut *isource*.

Le pilote associé au dictionnaire de définition d'une variable $x$ est finalement représentable dans la relation binaire

$$\mathbf{P_x}(\mathbf{jpagi}, jfpagi).$$

Le nombre de tuples de cette relation est majoré par NBFPAGI. Les valeurs de $jfpagi$ sont celles définies à l'annexe A. D'un point de vue pratique, la valeur d'une telle relation sera lue dans un fichier ou à travers une interface d'acquisition.

### Clé instrumentale

Une clé primaire instrumentale est introduite par numérotation séquentielle des valeurs lues. La variable *base* a été introduite pour fournir le point de départ de la numérotation: elle est initialisée à 1 par défaut mais une autre valeur peut être transmise en paramètre.

Cette clé instrumentale peut être supprimée ultérieurement après validation d'une autre clé : avec l'exemple du dictionnaire des communes, le couple (*departement*, *commune*) va remplacer le numéro d'ordre séquentiel après avoir vérifier que ce couple identifie effectivement un seul nom de lieu.

### Relations d'équivalence

Bien que les aspects liés au calcul des équivalences entre noms ne relève pas de cette étape, la méthode adoptée est évoquée ici dans la mesure où elle utilise l'attribut $jfpagi$ pour spécifier des relations d'équivalence.

La construction d'une relation d'équivalence à partir d'un pilote peut s'effectuer au moyen des valeurs de $fpagi$ non utilisées comme format PAGI, qui désignent alors pour l'utilisateur des plans de classement :

Un premier mode de construction définit préalablement des tables de chiffrement : nomenclatures de métiers, regroupement des graphies de patronymes phonétiquement semblables, classification éthymologique des toponymes, etc.

Un second mode plus complexe calcule les classes au moyen de fonctions dans une logique de système expert. Pour les toponymes et les patronymes, les transformations phonétiques usuelles sont formalisables

en tenant compte du contexte local à travers des fonctions définies sur les ensembles de phonèmes ; nasalisation, palatisation, chute de consomnes, diphtongaison, etc.

### 2.3.3 Fonctions relationnelles

Les relations qui contiennent les lexiques et dictionnaires ont un rôle fondamental dans les séquences de contrôle, d'apuration et de numérisation des données acquises. Les fonctions associées sont décrites globalement mais leur mise en œuvre relève d'une étape ultérieure.

#### Intégrité de référence

Les noms lus dans les actes appartiennent à des ensembles finis inclus dans $\mathcal{G}$ et gérés dans des dictionnaires. Ces dictionnaires sont construits par thésaurisation des noms nouveaux reconnus. La propriété selon laquelle tous les noms cités sont répertoriés dans les dictionnaires est appelée **intégrité de référence.**

La mise à jour des dictionnaires et des relations d'archivage s'effectue au cours d'une transaction pendant laquelle la base est inaccessible. La gestion transactionnelle garantit l'intégrité de référence, par validation à l'issue de la transaction ou annulation des modifications en cas d'incident.

Une telle transaction fait que la base est indéfiuie pour une période qui peut être longue dans la mesure où elle inclut le temps nécessaire pour la validation des nouveaux noms. La base est alors dans un état **verrouillé** qui peut durer plusieurs jours tant que les calculs définitifs ne sont pas validés.

#### Corrections

Dès lors que les clés primaires sont dûment définies, il est alors possible de construire des expressions relationnelles dont l'évaluation substitue aux valeurs erronées des relations d'archivage les valeurs corrigées dans les lexiques. Pour ce faire, la relation intermédiaire

$$\textbf{Correcteur}(\textbf{isource}, \textbf{iacte}, \textbf{irole}, \textbf{jtype}, \textit{hmauvais}, \textit{hbon})$$

est construite afin de pouvoir substituer aux valeurs erronées de l'attribut *hmauvais* les valeurs corrigées de l'attribut *hbon* dans les relations d'archivage. Cette substitution s'effectue dans une transaction au moyen d'expressions relationnelles dont l'évaluation peut consommer beaucoup de ressources si la cardinalité des ensembles opérandes est élevée.

### Numérisation

Les dictionnaires étant des ensembles finis, il est possible de remplacer les noms par un code numérique propre défini au moyen d'une fonction $f$ bijective : la fonction $f^{-1}$ permet de retrouver le nom à partir de la valeur numérique. Une telle fonction est gérée dans une relation binaire

$$\textbf{Chiffrement}(\mathbf{g}, n)$$

dans laquelle le nombre géré dans $n$ remplace le nom contenu dans $h$.

L'opération de numérisation par jointure naturelle remplace la relation définie sur $\mathcal{N}^4 \times \mathcal{G}$

$$\mathbf{A_1}(\textbf{isource}, \textbf{iacte}, \textbf{irole}, \textbf{jtype}, g)$$

par la relation définie sur $\mathcal{N}^5$

$$\mathbf{A_2}(\textbf{isource}, \textbf{iacte}, \textbf{irole}, \textbf{jtype}, n).$$

### Encryptage

Cette numérisation des noms propres offre un premier niveau de sécurité qui n'est pas nécessairement suffisant pour ouvrir la base pour des applications scientifiques externes. Un second niveau peut être introduit simplement au moyen d'une fonction d'encryptage de la valeur gérée dans l'attribut $n$.

## 2.4 Génération des références vers d'autres actes

Certaines valeurs lues dans un acte traduisent l'existence légale d'un autre acte et fournissent des informations pour le localiser dans l'espace et le temps, partiellement ou completement, de façon exacte ou sur une période estimée. Ces liens sont gérés sous la forme de clés numériques dites **références**.

### 2.4.1 Origine des références

L'existence d'inscriptions marginales, principalement dans les actes de naissance, signale l'existence d'un autre acte, avec normalement une date et un lieu d'établissement ainsi qu'un nom pour un mariage. Les reconnaissances d'enfants et les séparations consécutives à des divorces ou aux décès d'anciens conjoints, indiquées légalement dans le corps des actes de mariage, font également référence à d'autres actes explicitement cités.

Les dates explicites d'évènements de l'état civil relatifs à des personnes ainsi que les lieux d'origine localisent normalement des actes. Les lieux de résidence désignent les registres à dépouiller en priorité pour rechercher d'autres actes.

Des valeurs comme l'âge d'une personne et des expressions textuelles en usage présentes dans le corps de l'acte signalent implicitement l'existence d'un évènement sans toutefois le localiser précisément dans le temps et l'espace : mineur ou majeur, ici présent, décédé, veuf en première noce...

### 2.4.2 Représentation formelle des références

Les références vers d'autres actes sont des fonctions qui associent à un événement de l'état civil mentionné pour une personne citée dans un acte un rôle pour cette même personne dans un autre acte.

#### Relation de gestion des références

Les fonctions précédentes sont gérées dans la relation

**R(isource, iacte, irole, jtype, icible)**

dans laquelle

*isource* et *iacte* identifient l'acte d'origine,
*irole* contient le rôle de la personne dans cet acte,
*jtype* code l'évènement référencé et l'origine de la mention,
*icible* gère le rôle de la personne dans l'acte d'arrivée.

Le triplet (*isource*, *iacte*, *irole*) désigne une personne de l'acte d'origine : sujet, conjoint, enfant, parent.

Le couple (*jtype*, *icible*) identifie le fait civil référencé et permet de retrouver l'origine de la référence. L'attribut *jtype*, qui gère la variable *type*, est repris dans les relations d'archivage pour identifier le type d'information dans les classes PERSONNE, PERIODE et LIEU. La variable *cible*, gérée dans l'attribut *icible*, est calculable à partir de *type* : cet attribut est néanmoins conservé pour simplifier les calculs. Il est en effet plus efficace de rechercher les actes de mariage des hommes au moyen du prédicat

$$(icible = 2100)$$

que d'utiliser la formule ensembliste

$$(jtype \in \{20000, 20001, ...\}) \wedge (irole = 1100)$$

avec le risque d'incomplétude dans l'énumération.

### Séquencement des références

Les références associées aux mentions marginales sont repérées par une valeur positive de la variable *sequence* déduite de la valeur contenue dans *jtype*, comme décrit au paragraphe 1.4.2: ce numéro d'inscription, qui ne traduit pas nécessairement l'ordre chronologique des évènements, permet de différencier les actes de mariages ou divorces dont une personne est sujet.

Dans le contexte PAGI initial, une valeur nulle de la variable *sequence* signifie que la référence est générée dans le corps de l'acte. Or certains actes antérieurs sont parfois cités dans le corps de l'acte : mariage antérieur, décès d'un ancien conjoint, jugement de divorce, naissance d'un enfant. Le nombre de ces citations n'est pas déterminé a priori : les remariages après veuvage et les reconnaissances d'enfants sont relativement fréquents pour les périodes anciennes.

Cette situation peut être transcrite dans des listes qui regrouperaient un nombre variable de mentions marginales. Cette solution n'est pas utilisée dans le contexte PAGI actuel.

### Ordonnancement

Les références définissent à l'instant $t$ un ensemble $\mathcal{R}_t$.
Une première relation d'ordre notée

$$\mathcal{R}_{1t}(\mathbf{isource}; \mathbf{iacte}; \mathbf{irole}; \mathbf{jtype}; \mathbf{icible})$$

regroupe les références dans l'ordre des actes du registre, avec les mentions marginales de mariages puis divorces puis décès et ensuite les autres références.

La seconde relation d'ordre notée

$$\mathcal{R}_{2t}(\mathbf{icible}; \mathbf{irole}; \mathbf{isource}; \mathbf{iacte}; \mathbf{jtype})$$

définie un autre classement en fonction des valeurs des rôles dans l'acte référencé. Ce second plan va permettre de suivre par exemple les mariages et l'on pourra vérifier au vu des valeurs contenues respectivement dans *irole* et *jtype* que la référence provieut d'un acte de naissance.

### Génération des références

Si les dispositions légales aujourd'hui en vigueur pour l'état civil français sont effectivement appliquées, les inscriptions en marge de l'acte de naissance

d'une personne permettent de générer les références vers tous les actes dont cette personne est sujet.

De la même façon, toutes les références relatives aux deux sujets d'un acte de mariage pour des actes antérieurs dans lesquels ils sont également sujets doivent pouvoir être générées à l'issue de l'analyse des valeurs des variables de l'acte.

Pour les personnes citées autres que les sujets, les dates et lieux indiqués sont utilisables pour localiser d'autres actes dans le temps et l'espace.

### 2.4.3 Recherche des actes référencés

Les liens entre les actes sont calculés dans des expressions relationnelles dont l'évaluation en vraie grandeur nécessite une puissance de calcul importante. Nous décrivons ici une fonction qui associe à un rôle dans un acte d'un registre l'ensemble des rôles référencés dans les actes des registres.

Soit

$\gamma \in \mathcal{N}^3$ une valeur de la clé $(isource, iacte, irole)$,
$x \in$ H une variable d'identification d'une personne,
$\lambda$ la valeur de $x$ pour la clé $\gamma$,
$\mathcal{R}_\gamma$ l'ensemble des références associées à la clé $\gamma$,
$\widetilde{\mathcal{R}_\gamma}$ l'ensemble des clés $(isource, iacte, irole)$ référencés,
$\equiv$ une relation d'équivalence sur $\mathcal{G}$.

La recherche des actes dans lesquels $\lambda$ est sujet s'effectue en trois étapes.

1. La valeur $\lambda$ est construite en recherchant dans les relations $Nom$ et $Prenoms$ de la classe PERSONNE le nom et la liste de prénoms qui vérifient le prédicat $(isource, iacte, irole) = \gamma$.

2. L'ensemble $\mathcal{R}_\gamma$ est calculé par une extraction de la relation $R$ de tous les éléments qui vérifient le prédicat $(isource, iacte, irole) = \gamma$.

3. L'attribut *irole* contenant la variable *role* des relations d'archivage et l'attribut *icible* la variable *cible* du rôle référencé, l'ensemble $\widetilde{\mathcal{R}_\gamma}$ est calculé en recherchant dans les relations $Nom$ et $Prenoms$ les éléments qui vérifient les prédicats $(x \equiv \lambda)$ et $(role = cible)$.

# Chapitre 3

# Génération des relations

Il existe légalement en France, pour toute personne physique ayant vécu, un *acte initial* de naissance et un *acte final* de décès qui définissent un intervalle du temps calendaire. Sur cet intervalle peuvent se succéder chronologiquement des évènements qui donnent lieu à l'établissement d'autres actes : mariages, naissances d'enfants, séparations par décès du conjoint ou divorce.

Les informations contenues dans les actes sont généralement codifiées dans des variables définies empiriquement dans un univers d'acquisition des données : les fiches élaborées dans le contexte PAGI contiennent plus d'une centaine de variables. En l'absence de langage, les valeurs des variables sont des textes quelconques.

L'univers d'archivage des données décrit au paragraphe 1.5.2 propose un nombre restreint de variables définies sur $\mathcal{H}$ et réparties dans des classes. Ces variables codent des entités sémantiques jugées indissociables : noms et prénoms, dates, lieux complets, métiers, ...

Les contraintes de calcul conduisent à restreindre les domaines des attributs des relations aux ensembles $\mathcal{G}$ de noms et $\mathcal{N}$ des nombres naturels. En conséquence, les valeurs d'une variable complexe sont décomposées en noms et nombres et gérées pratiquement dans une ou deux relations, afin de prendre en compte les éventuels séquencements ou inclusions.

Ce chapitre décrit le mode de gestion de ces variables, quelles soient issues des actes d'un registre ou d'un lexique, en fonction de leurs classes d'appartenance. Il définit des relations de base à partir desquelles il est possible de construire d'autres relations pour les besoins applicatifs.

## 3.1 Mentions marginales

Les informations figurant en marge des actes de l'état civil sont des valeurs de variables des classes PERSONNE, PERIODE et LIEU définies postérieurement à la date d'établissement de l'acte.

### 3.1.1 Lecture des mentions et calcul des rôles associés

Les évènements associés aux meutions sont codés dans PAGI au moyen de variables instrumentales repérées par la valeur nulle de l'attribut *jclasse*. Chaque mention est repérée par un numéro d'ordre affecté au fur et à mesure de la lecture indépendamment de toute chronologie. Nous nous limitons dans le contexte de cette application au traitement des mentions marginales des actes de naissance.

Pour la période récente, les évènements de l'état civil relatifs à une personne sont normalement notés en marge de son acte de naissance, avec la date et le lieu de l'acte et le nom du conjoint pour les mariages. Les mentions de mariages, éventuellement multiples, font référence à quatre rôles :

*nouveau-né* dans l'acte de naissance,

*marié* ou *mariée* dans un acte de mariage,

*conjoint du nouveau-né* dans l'acte de naissance,

*mariée* ou *marié* dans un acte de mariage.

La mention de décès, unique et dernier élément de la suite chronologique des mentions, traduit l'existence de deux rôles :

*nouveau-né* dans l'acte de naissance,

*défunt* dans l'acte de décès.

Les mentions de divorces éventuellement présentes sont semblables à celles des mariages.

L'ordonnancement des directives de pilotage fait que ces variables sont traitées en premier. Elles sont conservées pendant le traitement de l'acte afin de pouvoir ensuite restituer les rôles pour les valeurs des variables des classes PERSONNE, PERIODE et LIEU lues dans la marge de l'acte.

### 3.1.2 Édition des références

Une référence est une valeur du quintuplet

$$(isource, iacte, irole, jtype, icible).$$

$(isource, iacte)$ identifie l'acte générateur de la référence,

*irole* code le rôle de la personne dans cet acte,

*jtype* reçoit le numéro d'ordre affecté à la mention,

*icible* contient le rôle dans l'acte référencé.

Une mention de mariage ou séparation génère deux références, l'une pour le nouveau-né et l'autre pour le conjoint cité, alors qu'une mention de décès en génère une seule.

## 3.2 Classe PERSONNE

La classe PERSONNE gère l'identité civile d'une personne, ses états matrimoniaux successifs et la légitimité de l'enfant.

### 3.2.1 Définition des variables

Les patronymes et les prénoms sont gérés dans deux variables textuelles distinctes définies sur l'ensemble $\mathcal{G}$, le sexe, l'état matrimonial et la légitimité étant codés dans des variables numériques.

#### Identification civile d'une personne

Toute personne physique est normalement identifiée dans un acte de l'état civil par son nom légal, ses prénoms et son sexe. Le nom légal figure parfois derrière le nom d'usage pour les femmes sous la forme « $x$ *née* $y$ ». Un surnom ayant valeur de nom d'usage peut aussi être noté. Tous ces noms définissent des patronymes gérés dans la variable *nom* : les ensembles de patronymes sont énumérés dans des dictionnaires.

Les prénoms successifs sont gérés dans la variable *prenom*, la valeur étant identifiée par le rang du prénom dans la liste transcrite. Les prénoms appartiennent également à un ensemble géré dans un dictionnaire.

La variable *genre* code le sexe comme indiqué au paragraphe 1.3.3.

### Etat matrimonial

L'état matrimonial d'une personne à une date donnée est défini au sens large comme la somme des trois variables

$$etat = matrimonial + genre + rang.$$

La variable *matrimonial* code précisément l'état matrimonial au sens de l'état civil. La vie civile d'une personne démarre avec son acte de naissance dans un état **célibataire** . L'acte de mariage définit un état matrimonial **marié**. L'acte de décès du conjoint fait passer la personne dans l'état **veuf** et l'acte de divorce la place dans l'état **divorcé** : :un nouveau mariage génère un nouvel état **marié**. Ces différents états sont codés ainsi :

*matrimonial* = 1000 : célibataire,
*matrimonial* = 2000 : marié,
*matrimonial* = 3000 : veuf,
*matrimonial* = 4000 : divorcé,
*matrimonial* = 9000 : information absente.

La variable *genre* est réintroduite dans le seul but de faciliter les traitements ultérieurs.

Le numéro d'ordre chronologique d'un mariage, veuvage ou divorce est lu s'il est effectivement transcrit et codé dans la variable *rang*, qui prend la valeur nulle par défaut.

### Légitimité d'un enfant

Tout enfant qui naît de parents mariés est déclaré **légitime** alors qu'un enfant de père inconnu est dit **naturel** et peut être ultérieurement **reconnu**. La légitimité est codée dans la variable *legitime*.

*legitime* = 0 : sans objet ou indéterminé,
*legitime* = 10 : enfant légitime,
*legitime* = 20 : enfant naturel,
*legitime* = 30 : enfant de parents inconnus,
*legitime* = 90 : information absente.

## 3.2.2 Syntaxe d'acquisition

Les variables de la classe PERSONNE utilisées dans le contexte PAGI d'acquisition des données sont définies par restriction de la syntaxe générale du

langage $H$. Ces restrictions sont décrites à partir de l'axiome $< hcitation >$ associé à la forme normale par les dérivations syntaxiques suivantes :

$< hcitation > \longrightarrow < personne >$
$< hcitation > \longrightarrow < lettre > [< chiffre]$
$< personne > \longrightarrow < patronyme > [, < prenoms >][(< liste >)]$
$< patronyme > \longrightarrow < homonyme > | < hnom >$
$< homonyme > \longrightarrow < repetiteur > [< marque >]$
$< marque > \longrightarrow < lettre > | < chiffre >$
$< prenoms > \longrightarrow < prenom > \{< blanc >< prenom >\}$
$< prenom > \longrightarrow < simple > [- < simple >]$
$< simple > \longrightarrow < nom > | < abreviation >$
$< abreviation > \longrightarrow < lettre > | < nom >$
$< liste > \longrightarrow < element > \{; < element >\}$
$< element > \longrightarrow [< gtag >:] < patronyme >$
$< element > \longrightarrow [< htag >:] < hgroupe >$
$< gtag > \longrightarrow < lettre >$
$< htag > \longrightarrow < lettre >$

La seconde règle permet de reconnaître les conventions PAGI actuelles de codage des variables *genre*, *matrimonial* et *legitime*, décrites en annexe A. Le chiffre optionnel traduit un numéro d'ordre géré dans la variable *rang* : l'expression « *Veuf en seconde noce* » lue dans l'acte est codée « *V2* » dans PAGI et génère ensuite la valeur 3102 pour la variable *etat*.

Le composant syntaxique $< homonyme >$ gère la répétition des patronymes des sujets de l'acte, le suffixe $< marque >$ n'étant utilisé que s'il est nécessaire de préciser le sujet concerné.

Les prénoms composés sont reconnus et les abréviations sont utilisables pour tous les prénoms simples.

Les préfixes $< gtag >$ et $< htag >$ introduisent respectivement des mots des ensembles $\mathcal{G}$ et $\mathcal{A}$ définis au paragraphe 1.1.4, les premiers représentant d'autres patronymes en usage pour une personne citée et les seconds des notes. Les dérivations reconnues sont indiquées en annexe A.

### 3.2.3 Identification des patronymes des sujets de l'acte

Les sujets de l'acte sont identifiées par des valeurs données de *isource* et *irole* et la propriété de préséance du sujet autorise la répétition d'un patronyme pour les valeurs dérivées de $< homonyme >$. Pour un mariage ou

un divorce, la lettre qui suit le répétiteur permet de préciser si le patronyme est celui de l'homme ou de la femme.

### 3.2.4 Normalisation de la transcription

La normalisation de la transcription des patronymes et des prénoms s'effectue en appliquant les règles suivantes :

- La première lettre d'un mot généré par la variable syntaxique $< nom >$ est transcrite sous la forme majuscule et les autres lettres sont éditées sous la forme minuscule, sauf pour les particules.

- Les particules sont des mots d'un ensemble énumérable utilisés dans la construction d'un patronyme et placés devant un autre mot. La particule peut être le premier mot d'un patronyme. Deux particules ne peuvent pas se suivre. Toutes les lettres des particules sont transcrites en minuscules.

- L'apostrophe permet habituellement l'élision d'une voyelle d'un article, sauf dans certaines transcriptions comme le *c'h* breton. Ces formes particulieres sont énumérées dans l'ensemble des fausses élisions propres aux patronymes, pour que la lettre qui suit l'apostrophe soit éditée sous la forme minuscule.

- Les abréviations sont des notations spécifiques définies préalablement par convention pour des prénoms dont l'usage est fréquent.Elles sont remplacées par le prénom associé. Les abréviations les plus communes des prénoms français sont *J* ou *J.* pour Jean, *M* ou *M.* pour Marie mais toute autre abréviation peut être intégrée.

### 3.2.5 Relations d'archivage

Les patronymes sont gérés dans première relation, une personne pouvant avoir plusieurs patrouymes en usage alors distingués par des valeurs distinctes du type d'information.

Les prénoms sont stockés dans une relation spécifique.

L'état matrimonial à la date de l'acte et la légitimité sont gérés dans deux autres relations.

### Identité civile

Le patronyme et le sexe sont gérés dans une première relation

**Nom(isource, iacte, irole , jtype**, *nom,sexe,nprenom,suite*).

L'attribut *jtype* permet de gérer les noms d'usage d'une personne : il reçoit la valeur de la variable

$$type = nature + sequence.$$

Il est ainsi possible de gérer plusieurs noms d'usage alors repérés par leur numéro d'ordre séquentiel dans la liste. La forme `patronyme()` étant étrangère au langage, une liste contient au moins un élément : la longueur de la liste est limitée à PATROMAX éléments.

L'attribut *nom*, défini sur $\mathcal{G}$ contient le patronyme.

Bien que le sexe soit déjà codé dans la variable *role* comme indiqué au paragraphe 1.3.3, la variable *genre* est reprise et gérée dans l'attribut numérique *sexe*.

Le nombre de prénoms est stocké dans *nprenom*.

L'attribut *suite* contient la valeur de la variable *rupture*.

### Liste des prénoms

Une seconde relation gère les prénoms en introduisant l'attribut supplémentaire *rang* qui précise le rang du prénom, soit

**Prenom(isource, iacte, irole, jtype, rang**, *nom*).

L'attribut *nom*, défini comme précédemment sur $\mathcal{G}$, contient le prenom. La liste des prénoms, est calculée par ordonnancement de l'ensemble des tuples qui vérifie le prédicat $(isource, iacte, irole, jtype) = \lambda$.

Les exceptions sont gérées dans la première relation, pour tout débordement ou précision, quelle qu'en soit l'origine.

### Sujets des actes

Une relation supplémentaire est créée dans le but d'établir un répertoire des sujets de tous les actes. Un sujet est désigné seulement par son nom et le prénom qui suit, dit premier prénom. L'attribut *sexe* est introduit pour pouvoir distinguer mari et femme pour les mariages. Ces données sont gérées dans la relation

**SujetActe(isource, iacte, sexe**, *nom*, *prenom*),

utilisable pour localiser des ensembles d'actes pour des patronymes.

#### État matrimonial

L'état matrimonial lu dans l'acte est géré dans l'attribut numérique *matrimonial* de la relation

$$\mathbf{Matri(isource, iacte, irole, jtype}, \mathit{matrimonial}).$$

#### Légitimité

La relation

$$\mathbf{Legitime(isource, iacte, irole, jtype}, \mathit{leg})$$

est utilisée uniquement pour gérer des situations dans laquelle la non légitimité est établie ou possible, principalement pour le nouveau-né.

L'attribut *leg* contient la valeur de la variable *legitime*, codée explicitement en amont après lecture d'expressions comme « *... a accouché de Joséphine Alphonsine de père inconnu ...* » ou « *... enfant déposé au grand berceau de l'Hotel-Dieu et confié à la garde et aux soins de ...* ».

### 3.2.6 Dictionnaires des patronymes et prénoms

Les dictionnaires sont construits à partir de la projection relationnelle de l'attribut *nom* des relations d'archivage *Nom* et *Prenom*.

#### Numérisation des patronymes

La numérisation des noms est réalisée au moyen d'une fonction $f$ bijective définie de $\mathcal{G}$ vers $\mathcal{N}$. La fonction réciproque $f^{-1}$ de $\mathcal{N}$ vers $\mathcal{G}$ restitue le nom à partir du code numérique. Ces fonctions sont gérés dans une relation binaire munie de deux clé primaires notées

$$\mathbf{D(nom}, n) \text{ et } \mathbf{D}(\mathit{nom}, \mathbf{n}),$$

l'attribut $n$ recevant le numéro associé au nom.

Pour des raisons d'efficacité, la longueur des patronymes peut être restreinte à LPATRONYME caractères au plus, la constante étant choisie de telle sorte qu'il n'existe pas deux noms pour lesquels les LPATRONYME premiers caractères seraient identiques : les noms plus long sont alors traités comme dans les actes, ce qui nécessite l'ajout de l'attribut *suite*.

Une fonction de numérisation simple reprend la relation d'ordre associé à un classement alphabétique, la numérotation s'effectuant à partir d'une base transmise en paramètre. La validation s'effectue dans une transaction à l'issue de laquelle les propriétés d'unicité des deux clés primaires sont dûment vérifiées.

#### Gestion des équivalences

Le dictionnaire des patronymes peut être complété en introduisant les attributs de gestion des classes d'équivalence définies au paragraphe 1.2.3.

## 3.3 Classe PERIODE

La classe PERIODE gère les dates et les indications d'âge utiles pour situer la date d'un évènement dans un intervalle de temps. L'évènement, associé à une personne citée, est codé pour les dates dans un composant PAGI spécifique de la classe PERIODE et est implicitement une naissance pour un âge.

Une date est traduite dans le calendrier grégorien sous la forme d'un triplet $(jour, mois, an)$.

Un âge est une grandeur numérique notée dans la variable *age* et relative à la date d'établissement de l'acte. L'unité de mesure effective est indiquée dans les variables *unite* et *forme* décrites au paragraphe 1.4.2.

Les valeurs gérées dans cette classe sont numériques : les dictionnaires sont sans objet.

Le mécanisme de gestion des exceptions demeure utilisable.

### 3.3.1 Syntaxe d'acquisition

Les variables de la classe PERIODE sont définies au moyen des variables syntaxiques intermédiaires suivantes :

$< hcitation > \longrightarrow < periode >$
$< periode > \longrightarrow < date > [< hinclusion >]$
$< periode > \longrightarrow < jour > | < mois > | < annee >$
$< periode > \longrightarrow < age > | < indication > | < delai >$
$< date > \longrightarrow < jour > [< sdate >< mois > [< sdate >< an >]]$
$< jour > \longrightarrow < nombre >$
$< sdate > \longrightarrow -|/| < blanc >$

$< mois > \longrightarrow < nombre > | < nom > | < chiffre >< nom >$
$< annee > \longrightarrow < an > [< hinclusion >]$
$< an > \longrightarrow < nombre > |\text{AN}< chiffre >< chiffre >$
$< age > \longrightarrow < nombre > | < uage > \{< sdate >< uage >\}$
$< uage > \longrightarrow < nombre >< unite >$
$< unite > \longrightarrow a|m|s|j|h$
$< indication > \longrightarrow < hnom >$
$< delai > \longrightarrow < nombre >$

Les composants d'une date sont séparés par un trait d'union, une barre oblique ou un espace.

Les mois sont représentés par un numéro ou un nom d'une liste prédéfinie.

Les dérivations de la variable syntaxique < *indication* > sont des mots et expressions utilisés dans un contexte national et interprétés pour localiser la date de naissance d'une personne vivante ou la date de décès d'un défunt : *mineur*, *majeur*, *présent*, *comparu*, *vivant*, *défunt*, *feu*, *mort-né*...

Le calendrier républicain est reconnu à travers une dérivation syntaxique spécifique qui génère le préfixe AN suivi des deux chiffres de l'année républicaine, comme AN02.

La variable syntaxique < *uage* > est introduite pour introduire plusieurs unités de mesure du temps pour un âge.

### 3.3.2 Relations d'archivage

Une date relative à une personne citée est liée à un évènement. En conséquence, son traitement doit générer la référence vers l'acte correspondant.

L'âge d'une personne permet de situer la naissance d'une personne dans un intervalle de temps borné supérieurement par la date de l'acte, donc génère une référence vers l'acte associé à cette naissance. Si la personne est décédée, une seconde référence est générée vers un acte de décès, à rechercher dans un autre intervalle de temps.

Les relations de gestion des dates et âges sont composées d'attributs numériques.

### Dates des évènements

Les dates des évènements sont gérées dans la relation

**DateFait**(**isource**, **iacte**, **irole**, **jtype**, *jour*, *mois*, *an*).

L'attribut *jtype* gère simultanément

l'évènement associé, lu dans le pilote pour les sujets ou calculé après lecture du type de date dans l'environnement PAGI ;

le mode d'élaboration de la date, lue directement ou calculée à partir du délai entre l'évènement et la date d'établissement de l'acte ;

la précision de la date, qui peut être exacte ou partiellement indéterminée si le délai n'est pas précisé pour les naissances et décès ;

le calendrier d'origine.

Les indéterminations sont représentées par une valeur nulle et les exceptions ne sont pas admises.

L'ordonnancement des directives de pilotage décrit au paragraphe 2.1 fait que les dates associées aux sujets sont calculées en premier et peuvent ainsi être conservées pendant le traitement de l'acte. Elles sont utilisées pour réaliser les contrôles chronologiques et pour déterminer les intervalles de temps à partir des indications d'âges des personnes citées.

### Dates des actes

Les dates des actes sont associées aux sujets dans les instructions de pilotage. Elles diffèrent généralement des dates des évènements pour les naissances et décès , avec un délai de déclaration normalement restreint à quelques jours. Elles sont conservées dans la relation

**DateActe**(**isource**, **iacte**, *jour*, *mois*, *an*)

Cette relation est définie et utilisée pour localiser les ensembles d'actes disponibles sur une période historique.

### Âges

La relation de gestion des âges

**Age**(**isource**, **iacte**, **irole**, **jtype**, *age*, *suite*, *jour*, *mois*, *an*)

contient la date d'établissement de l'acte et la valeur de l'âge dans le système d'unités codé dans l'attribut *jtype*.

Si l'âge ne figure pas en clair, les expressions textuelles sont remplacées par des valeurs numériques constantes déterminées en fonction des usages et de critères démographiques, mais aussi après examen de distributions statistiqnes observées pour évaluer un risque d'erreur.

En effet, pour des raisons d'efficacité, il n'est pas possible de majorer ces constantes. Le fait d'augmenter la longévité maximale augmente la probabilité pour que la date de naissance d'une personne soit effectivement dans l'intervalle déduit de l'âge, mais en contrepartie, cet intervalle est très large. En réduisant cette valeur maximale, la date de naissance est localisée plus précisément dans la plupart des cas mais peut parfois se situer hors intervalle. En conséquence, il convient de caractériser les constantes choisies par une mesure d'un risque d'erreur associé.

La valeur de la variable *age* déduite des indications présentes est finalement une fonction numérique de plusieurs variables. Dans ce contexte applicatif, la fouction

$$\psi(evenement, fonction, sexe, forme, an)$$

paraît suffisante pour évaluer l'âge. L'année est introduite principalement pour tenir compte de dates à partir desquelles une définition légale a pu être modifiée, par exemple la majorité civile.

Vu les domaines de définition effectifs de toutes variables, il est possible de définir la fonction $\psi$ par énumération des sexuplets

$$(age, evenement, fonction, sexe, forme, an).$$

### 3.3.3 Génération des références

La génération des références s'effectue immédiatement pour les dates et les âges lus dans le corps de l'acte, les références associées aux mentions marginales étant générées lors de la reconnaissance des évènements. Les valeurs gérées dans les attributs *isource*, *iacte*, *irole* et *jtype* sont celles des relation d'archivage. L'attribut *icible*, qui code le rôle dans l'acte référencé, reçoit la somme des variables *fonction* et *genre* définies au paragraphe 1.3.3, la valeur de *fonction* étant déduite de la nature de l'évènement.

L'environnement PAGI actuel n'autorisant qu'une seule date pour une personne citée, la variable *rang*, définie dans le pilote, peut rester nulle. Cela se traduit par un nombre nul unités pour la variable *type*, alors que

ce nombre est par construction positif pour les mentions marginales. Il est néanmoins possible de prévoir plusieurs dates distinguées dans le pilote par des valeurs différentes du type d'information, comme pour les mentions marginales.

L'origine de la référence est notée dans la variable *nature* dont la valeur se déduit de celle de *type*, comme indiqué au paragraphe 1.4.2.

## 3.4 Classe LIEU

La classe LIEU gère les formes toponymiques transcrites dans les actes, des codes associés aux entités administratives d'un territoire national et les adresses postales.

Pour les données françaises, le lieu d'établissement de l'acte est la commune après la Révolution et la paroisse antérieurement. Les communes sont rattachées administrativement à un département et les paroisses à un diocèse. L'adresse des personnes citées est souvent un toponyme qui désigne une partie du territoire administratif du lieu d'enregistement, hameau, écart ou bourg en zone rurale, quartier en zone urbaine.

### 3.4.1 Définition des variables

Les toponymes sont définis sur l'ensemble $\mathcal{G}$, les adresses postales sur $\mathcal{A}$ et les codes utilisés sont numériques.

#### Identification textuelle

L'identification complète d'un lieu de résidence pour une personne citée dans un acte s'effectue selon un protocole qui sépare plusieurs valeurs textuelles associées à des variables qui identifient des parties de territoire imbriquées. Pour les registres civils, ces valeurs sont énumérées dans une suite qui peut contenir successivement le département, l'arrondissement, le canton, la commune et finalement un toponyme désignant un écart ou l'adresse précise dans une ville.

De cette structure hiérarchique constituée de toponymes reliés par des conjonctions sont extraites des valeurs textuelles gérées dans les variables *cheflieu*, *division* et *adresse*.

1. *cheflieu*, définie sur l'ensemble $\mathcal{G}$, contient le nom du lieu d'établissement des actes, commune ou paroisse selon la période. Ces toponymes

définis pour un pays et une période historique définissent un premier ensemble $\mathcal{G}_1$ : une paroisse de l'Ancien Régime appartient à $\mathcal{G}_1$.

2. *division* également définie sur l'ensemble $\mathcal{G}$ reçoit un toponyme qui désigne une subdivision rattachée au chef-lieu : hameau, écart, appellation *bourg* pour les communes rurales, quartier, ancienne paroisse. Ces toponymes qui désignent une portion de territoire dans un contexte géographique local définissent un second ensemble $\mathcal{G}_2$ : une ancienne paroisse citée dans le corps d'un acte civil est une subdivision de la commune qui appartient à $\mathcal{G}_2$, alors que cette même paroisse appartient aussi à $\mathcal{G}_1$ pour les actes de l'Ancien Régime. Plusieurs subdivisions, imbriquées ou chevauchantes, peuvent être citées.

3. *adresse*, définie sur l'ensemble $\mathcal{A}$, reçoit les adresses postales. Ces adresses sont des expressions alphanumériques qui désignent un cheminement vers un logement sans nécessairement préciser le toponyme qui désigne la portion de territoire sur laquelle est implanté ce logement. Une adresse est unique.

La normalisation de la transcription graphique et les calculs sont réalisés uniquement sur les toponymes des variables *cheflieu* et *division*. Les adresses sont seulement conservées pour les étapes suivantes de traitement.

En pratique, l'identification par les toponymes est ambiguë. Avec le dictionnaire officiel des communes actuel, l'ambiguïté est levée en associant le département mais celà ne concerne que la période récente. En ce qui concerne les hameaux, les noms sont indissociables du lieu auquel ils sont rattachés et le couple de toponymes n'est pas suffisant pour localiser de façon certaine une portion de territoire.[1]

Du point de vue théorique, celà signifie que le domaine défini par le produit cartésien $\mathcal{G}_1 \times \mathcal{G}_2$ ne permet pas d'identifier un lieu.

### Identification numérique

Les noms des départements français sont immédiatement remplacés par le code numérique du département noté dans la variable *departement*. Les anciennes paroisses peuvent pour la plupart être rattachées à un département actuel.

[1] Cette situation peut se produire lorsqu'un commune ou paroisse regroupe des territoires antérieurement séparés. Un exemple illustratif concerne les des deux hameaux de *Livriac* situés sur la commune d'*Iffendic* en *Ille-et-Vilaine*, distants de quelques kilomètres seulement mais situés de part et d'autre de la frontière entre deux cités gallo-romaines.

Le remplacement des noms de communes par les numéros officiels actuels plus complexe et a fait l'objet de travaux spécifiques.[2] L'environnement PAGI utilise ce code commune pour réaliser cette codification en amont : il est donc repris dans la variable *commune*.

Les noms des subdivisions citées dans une commune sont aussi à remplacer par un code préalablement défini. La variable *section* est introduite pour recevoir ces codes, qui sont à définir dans une étape ultérieure avec les données PAGI actuelles. Pour d'autres acquisitions, ils serait vraisemblablement possible d'introduire directement ces codes à partir de la liste des lieux habités connus de la commune, les problèmes d'homonymies étant réglées au moyen de conventions simples liées aux usages, par exemple une orientation par rapport au chef-lieu ou une limite naturelle.[3].

Les subdivisions font référence à différentes partitions du territoire : ancienne paroisse, chapelle, hameau, écart, etc. La nature de ces subdivisions est codée dans l'attribut *jtype* comme indiqué au paragraphe 1.4.2.

#### Graphie des toponymes

Un toponyme a valeur d'identifiant phonétique d'un territoire. Le mode de transcription graphique de cet identifiant dépend de multiples usages dans des contextes géographiques, historiques et sociaux diversifiés. Il en résulte l'existence de plusieurs graphies pour un même toponyme : l'attribut numérique *graphie* est introduit pour gérer ces variantes de transcription.

### 3.4.2 Syntaxe d'acquisition

La notion d'inclusion des territoires repérés par des toponymes est traduite dans le langage $H$ par la forme syntaxique globale

$$< groupe > (< inclusion >).$$

Si la dérivation de $< groupe >$ produit le toponyme qui désigne le lieu de l'acte, on ne sait pas a priori si la dérivation de $< inclusion >$ génère un toponyme, une adresse, une autre subdivision administrative ou encore une note à caractère exceptionnel. Pour résoudre cette ambiguïté, un préfixe optionnel a été introduit pour pouvoir discriminer explicitement les cas dans

[2] **Abdellatif Menkassi**, *Conception et réalisation d'une application graphique pour la gestion des toponymes*, INRA, 1995.

[3] Avec l'exemple de la note précédente, il serait nécessaire de distinguer *Livriac Nord* et *Livriac Sud*.

la phase d'acquisition des données, ce qui est traduit formellement par les dérivations syntaxiques suivantes :

$< hcitation > \longrightarrow < cheflieu > [(< liste >)]$
$< liste > \longrightarrow < element > \{; < element >\}$
$< cheflieu > \longrightarrow [< gtag >:] < toponyme > | =$
$< element > \longrightarrow [< gtag >:] < toponyme >$
$< element > \longrightarrow [< htag >:] < hgroupe >$
$< toponyme > \longrightarrow < hnom >$
$< gtag > \longrightarrow < lettre >$
$< htag > \longrightarrow < lettre >$

Les composants syntaxiques ont les fonctions suivantes :

$< cheflieu >$ génère le toponyme d'un lieu d'établissement des actes : le composant optionnel $< gtag >$ est utilisé pour préciser la nature du lieu, commune ou paroisse exclusivement.

$=$ permet de répéter le lieu de l'acte, connu au début du traitement de par l'ordonnancement des directives de pilotage.

$< element >$ produit une expression alphanumérique préfixée par une marque.

$< gtag >$ et $< htag$ introduisent respectivement des toponymes définis sur $\mathcal{G}$ et des adresses ou notes définies sur $\mathcal{A}$.

D'autres règles syntaxiques propres à l'environnement PAGI ont été introduites pour pouvoir reprendre une codification déjà réalisée : elles sont décrites en annexe A.

### 3.4.3 Normalisation

La normalisation de la transcription des toponymes affectés dans les variables *cheflieu* et *division* est effectuée en appliquant les règles suivantes :

- La première lettre d'un mot d'un toponyme est transcrite sous la forme majuscule sauf pour les prépositions ; les autres lettres sont éditées sous la forme minuscule.

- Les prépositions sont des mots d'un ensemble fini utilisés dans la construction des toponymes pour relier deux noms propres : toutes les lettres sont en minuscules.

- L'apostrophe est parfois utilisée, comme pour les patronymes, pour représenter des fausses élisions, énumérées dans un ensemble propre aux toponymes : la lettre qui suit l'apostrophe est, seulement dans ce cas, éditée sous la forme minuscule.

- Les abréviations sont des notations définies préalablement par convention pour désigner des noms propres fréquents dans les toponymes. Elles sont remplacées par les noms associés dans le toponyme édité : *St* et *Ste* sont des abréviations usuelles de **Saint** et **Sainte** dans les toponymes français.

- La barre oblique (/) est un séparateur dans un toponyme qui remplace usuellement la préposition **sur**.

- Le blanc séparateur entre les noms, prépositions et articles d'un toponyme est remplacé par un tiret (-), sauf si cet espace est situé entre l'article et le premier nom propre de ce toponyme.[4] Les articles sont les mots d'un ensemble fini.

### 3.4.4 Relations d'archivage

Les toponymes sont gérés dans l'attribut *toponyme*, défini sur le domaine $\mathcal{G}$, dans deux relations distinctes, l'une pour les chefs-lieux administratifs et l'autre pour les subdivisions : le type de lieu est noté dans l'attribut *jtype*. Ces relations sont utilisables comme opérandes dans les expressions relationnelles construites pour automatiser les traitements.

Les adresses sont conservées dans l'attribut *adresse*, défini cette fois sur le domaine $\mathcal{A}$, dans une troisième relation. La variabilité des formes graphiques autorisées et l'absence de normalisation font que cette relation est difficilement utilisable dans des séquences automatisées de calcul.

Les notes sont traitées avec les exceptions. Les troncatures sont effectuées pour tout toponyme de plus de LTOPONYME caractères.

[4] Il existe avec la convention INSEE quelques exceptions à cette règle qui sont gérées avec les exceptions.

### Chefs-lieux

Les toponymes des lieux d'établissement des actes, dits chefs-lieux, sont gérés dans la relation

**ChefLieu**(**isource,iacte,irole,jtype**,*departement,commune,graphie, toponyme,ltoponyme,suite*).

L'attribut *jtype* reçoit la valeur de la variable

$$type = evenement + nature + cause.$$

Les chefs-lieux sont exclusivement des communes ou paroisses : la nature du lieu est précisée dans le pilote, codée dans l'acte au moyen d'un préfixe optionnel ou reste indéterminée. Les variables *evenement* et *cause* sont définie dans le pilote. La variable *sequence* est sans objet pour les chefs-lieux.

Les attributs *departement* et *commune* contiennent les codes numériques des départements et communes auxquels est actuellement rattaché le chef-lieu. L'attribut *graphie* est introduit pour pouvoir coder ensuite au vu des dictionnaires la graphie effectivement utilisée dans l'acte : il est donc indéterminé dans la phase d'acquisition des données.

Le toponyme est géré dans l'attribut textuel *toponyme* et sa longueur est conservée dans *ltoponyme*, dans un but d'amélioration de l'efficacité des calculs ultérieurs.

La distinction entre chef-lieu et subdivision étant faite par analyse syntaxique, les ambiguïtés dans le contenu de l'acte peuvent se traduire par des erreurs : « `Le Verger` » désigne un chef-lieu alors que « `=(Le Verger)` » correspond à un hameau rattaché au lieu d'établissement de l'acte. La recherche de telles erreurs pourra s'effectuer par examen des situations pour lesquelles il n'existe pas de subdivision.

### Lieux des actes

Les lieux des actes, associés aux sujets dans les instructions de pilotage, sont gérés dans la relation

**LieuActe**(**isource**, **iacte**, *departement*, *commune*, *graphie*, *toponyme*),

utilisable pour localiser les ensembles d'actes disponibles sur une région. C'est en fait un sous-ensemble de la valeur de *ChefLieu*.

### Subdivisions

Les toponymes qui désignent des lieux habités rattachés à un chef-lieu sont gérés dans la relation

**Division**(**isource,iacte,irole,jtype**,*departement,commune,section,graphie toponyme,ltoponyme*).

L'attribut *jtype* reçoit la valeur de la variable

$$type = evenement + nature + cause + sequence.$$

La valeur de la variable *nature* est déduite de la dérivation de $< gtag >$.

La variable *sequence* gère le rang des éléments de la liste des subdivisions. La forme `toponyme()` étant étrangère au langage, une liste contient au moins un élément. La longueur de la liste est limitée à TOPOMAX éléments.

Les attributs *departement* et *commune* sont réintroduits pour simplifier les calculs, bien que les valeurs soient gérées une première fois dans la relation *ChefLieu*. L'attribut *section* est un code numérique affecté après énumération des subdivisions existantes du chef-lieu. Il est défini avec les codes des départements et communes dans les dictionnaires. Il peut être inconnu lors de la lecture des actes et prend alors la valeur nulle par défaut.

Les éventuelles exceptions sont gérées dans la relation *ChefLieu*.

### Adresses

Les adresses des habitations rattachées à un chef-lieu sont énumérées dans la relation

**Adresse**(**isource,iacte,irole,jtype**,*adresse, suite*).

La valeur de *type* est calculée de la même façon que pour les subdivisions.

L'attribut *suite* gère l'exception levée pour une adresse dont la longueur serait supérieure à LADRESSE.

### Construction d'autres relations

La clé primaire commune permet de produire des relations plus complexes en fonction des besoins. Chefs-lieux et subdivisions peuvent ainsi être vus dans une même relation.

### 3.4.5 Dictionnaires des toponymes

Les toponymes sont définis à plusieurs niveaux géographiques quant à l'étendue du territoire circonscrit. Pour une approche démographique, le niveau le plus fin est le hameau qui regroupe plusieurs habitations, voire l'écart réduit à une seule maison isolée. Une approche patrimoniale pourrait utiliser une partition plus fine avec des microtoponymes parcellaires.

Les dictionnaires des lieux d'établissement des actes sont initialisés à partir des dictionnaires officiels des communes disponibles puis complétés par ajout des noms en usage dans les documents pour les périodes plus anciennes.

Les hameaux et écarts sont administrativement rattachés à un chef-lieu.

#### Dictionnaire des lieux d'établissement des actes

Les lieux d'établissement des actes sont définis sur la base d'une partition du territoire français établie à une date précise et gérés dans la relation

**Commune**(*nom*, **jtype, departement,commune,graphie**).

Le code officiel du lieu est le couple (*departement*, *commune*).

L'attribut *jtype* gère la variable *nature* utilisée pour localiser les anciennes communes ou paroisse. Les exemples suivants pris en Ille-et-Vilaine illustrent les cas de figure rencontrés, la valeur de *nature* étant notée entre parenthèses :

`Saint-Servan-sur-Mer`(100) : commune rattachée à Saint-Malo ;
`Coulon`(200) : paroisse rattachée à Montfort après la Révolution ;
`Iffendic`(100) : commune actuelle ;
`Saint-Barthélémy`(300) : non autorisé, un acte n'étant pas établi dans un hameau.

Le triplet (*jtype*, *departement*, *commune*) est suffisant pour identifier le lieu d'établissement d'un acte.

#### Dictionnaire des subdivisions

Les subdivisions sont gérées dans la relation

**Division**(**nom**, *jtype,departement,commune,section,graphie*).

Ce dictionnaire est complété à partir des lexiques construits par énumération des valeurs lues dans les documents.

### 3.4.6 Génération des références

De la même façon que pour les dates, il est possible d'utiliser un composant PAGI spécifique dans la classe LIEU pour coder l'évènement associé. L'environnement PAGI actuel n'utilise pas cette possibilité et code dans des variables séparées les lieux d'origine et de résidence, comme indications pour la recherche d'évènements de l'état civil. Ces deux types de lieux sont repérés par les valeurs de la variable *nature* gérée dans l'attribut *jtype* du pilote.

Les attributs *isource*, *iacte*, *irole* et *jtype* reçoivent les valeurs définies pour les relations d'archivage. Pour les lieux de résidence ou d'origine, la variable *fonction* reçoit la valeur nulle afin de pouvoir rechercher tout évènement de l'état civil qui serait survenu en ces lieux. Pour les évènements explicites, sa valeur est déduite de la variable *evenement*. L'attribut *icible* reçoitla somme des variables *fonction* et *genre*.

Les références à partir des lieux sont exclusivement générées pour les valeurs contenues dans le corps de l'acte. La variable *rang* est utilisable pour mettre en œuvre à partir du pilote différentes stratégies de codage comme pour les dates.

## 3.5 Classe STATUT

La classe STATUT regroupe des variables descriptives du statut social de la personne au moment où elle est citée. Ces descriptions font référence principalement à des ensembles énumérés de noms respectivement pour des métiers, des fonctions collectives, des titres, des diplômes, etc. L'interprétation des noms pour les affecter dans une variable est souvent ambiguë: *docteur* désigne un métier ou un titre sanctionné par un diplôme alors que *maréchal* peut correspondre à un grade militaire ou un métier.

D'autres variables sont définies par un nombre restreint de modalités, comme la présence ou l'absence de signature, vue comme un indicateur et non comme une marque d'identification de la personne.

Les valeurs de ces variables sont archivées sous la forme normale du paragraphe 1.1.4 et consultables pour examiner si deux citations apparemment équivalentes dans des actes différents concernent effectivement la même personne. Elles seront traitées dans une étape ultérieure en relation avec les dictionnaires de définition associés aux variables effectivement présentes.

### 3.5.1 Syntaxe

Les variables de la classe STATUT sont produites au moyen des règles :

$< hcitation > \longrightarrow < liste > | < lettre >$
$< liste > \longrightarrow < element > \{< delimiteur >< element >\}$
$< element > \longrightarrow < hnom > [< hinclusion >]$

Il s'agit pratiquement de listes de noms ou de codes simples. La variable syntaxique $< hinclusion >$ permet seulement de noter des précisions à caractère exceptionnel.

### 3.5.2 Relations d'archivage

Les noms sont lus dans des listes gérées telles quelles et le seul code retenu pour les actes de l'état civil traite de la signature.

#### Listes

L'ensemble des noms, quelle que soit la variable auxquels ils se rapportent, sont gérés dans la relation

**Social(isource,iacte,irole,jtype,***nom,lnom,suite***)**.

L'attribut *jtype* reçoit la valeur de la variable *sequence* qui code le rang de l'élément dans la liste. L'attribut *nom* contient le nom, *lnom* sa longueur.

Les exceptions sont levées pour des noms de plus de LSTATUT caractères ou si une expression parenthèsée introduit une précision.

#### Signature

Les parents, témoins et mariés ont la possibilité de signer les actes, mais ils peuvent déclarer ne pas savoir faire : cette information est gérée dans la variable *signature* avec la convention

*signature* = 0 : indéterminé,
*signature* = 10 : a signé,
*signature* = 20 : a déclaré ne pas savoir signer,
*signature* = 30 : a signé d'une croix,
*signature* = 40 : absent,
*signature* = 90 : confirmation d'absence de cette information.

La variable *signature* est gérée dans la relation

**Signature(isource,iacte,irole,jtype,***signature***)**.

### 3.5.3 Dictionnaires

Les dictionnaires sont utilisables pour répertorier les termes en usage pour désigner les métiers et les différentes activités. Ils sont gérés dans la relation

**Actif**(*nom*, **jtype, plan, graphie**).

L'attributs *jtype* contient le numéro du plan de classement et l'attribut *plan* sa valeur.

Comme pour les toponymes, l'attribut *graphie* est nécessaire pour distinguer des métiers équivalents mais transcrits différemment selon les lieux et les périodes.

## 3.6 Classe PARENT

La classe PARENT permet de gérer la nature des liens, familiaux ou autres, entre des personnes citées et les sujets de l'acte, voire entre les sujets.

Ces liens sont souvent explicités au moyen de constuctions usuelles dans lesquelles deux noms sont reliés par une préposition: l'expression « frère **de** la mariée » traduit un lien de parenté connu avec un sujet de l'acte. Plusieurs liens avec un sujet peuvent être cités pour une même personne.

### 3.6.1 Syntaxe

Les valeurs des valeurs des variables de la classe PARENT sont structurées au moyen des délimiteurs définis pour le langage $H$ : les deux points remplacent la préposition de l'exemple précédent et le point-virgule sépare les éléments de la liste de liens. Cette convention est formellement traduite par les dérivations

$< hcitation > \longrightarrow < lien >\{; < lien >\}$
$< lien > \longrightarrow < hnom > [: < marque >][< hinclusion >]$

Le composant syntaxique $< hinclusion >$ n'est utilisé que pour gérer des exceptions.

### 3.6.2 Relation d'archivage

La variable *genre* code le sexe du sujet avec lequel est établi le lien : sa valeur doit nécessairement être définie pour les actes de mariage et les jugements de divorce.

La variable *rang* contient le rang du lien.

L'attribut *jtype* reçoit dans ce contexte la valeur *genre* + *rang*.

Le nom simple ou complexe désignant la relation avec le sujet, situé à gauche de la préposition dans la construction précédente, est conservé tel quel dans l'attribut *nom*, afin de pouvoir différer dans une étape ultérieure le traitement d'expressions en usage dont l'interprétation peut dépendre du contexte régional, comme l'« oncle à la mode de Bretagne **de** la femme ».

Les liens sont finalement archivés dans la relation

**Parent(isource,iacte,irole,jtype**,*nom,suite*).

Les exceptions sont levées pour des noms de plus de LPARENT caractères ou si une expression parenthèsée introduit une précision.

### 3.6.3 Dictionnaires

Les expressions en usage désignant le lien de parenté relativement à un sujet sont répertoriés dans un dictionnaire géré dans la relation

**Famille**(*nom*, **degre, lien, graphie**).

Les attributs *degre* rt *lien* code respectivement le degré de parenté et la nature du lien.

## 3.7 Classe MESURE

Les variables de la classe MESURE ont été introduites pour prendre en compte la mesure de grandeurs entières ou décimales faites dans un système d'unités. Il s'agit d'un nombre suivi d'un terme appartenant à un lexique d'unités possibles, ce qui se traduit formellement par la syntaxe

$< hcitation > \longrightarrow < decimal > [< unite >]$
$< decimal > \longrightarrow < nombre > . < nombre > [< unite >]$
$< unite > \longrightarrow [< blanc >< nom >$

Les mesures sont gérées dans la relation d'archivage

**Mesure(isource,iacte, irole, jtype**,*mesure*).

Le système d'unités est codée dans l'attribut *jtype*.

## 3.8 Classe CITATION

La classe < *citation* > permet de gérer la transcription normalisée de valeurs textuelles générées par le langage $H$, sans modification, et sans restriction quant à la longueur du texte.

Les textes sont gérés en utilisant les attributs *k*, *rupture* et *hsegment* définis pour les exceptions au paragraphe 2.2.1, dans la relation

**Citation(isource,iacte, irole, jtype,k**,*rupture,hsegment*).

L'attribut *irole* permet de cibler l'information vers une personne citée et *jtype* de classer ces citations selon des critères définis dans le pilote. Aucune exception ne peut être levée dans cette classe.

## 3.9 Classe GESTION

Les variables de la classe GESTION gèrent la clé numérique d'identification des actes et des plans de classement des actes définis sur l'ensemble $\mathcal{N}$.

La valeur LACLE de la variable *jfpagi* dn pilote signifie que le composant PAGI correspondant contient la clé effective, qui sera donc lue dans les actes. Si cette valeur n'existe pas, c'est le numéro d'ordre séquentiel de l'acte qui est retenu par défaut.

Les autres valenrs non nulles de *jfpagi* sont utilisables pour gérer des plans de classement. Il suffit pour celà de déclarer des composants PAGI particuliers qui contiennent des nombres décimaux, par exemple les codes commune et département des lieux de naissance des sujets.

Les plans de classement sont stockés dans la relation

**Gestion(isource,iacte,irole, jtype**,*plan*).

L'attributs *jtype* contient le numéro du plan de classement et l'attribut *plan* sa valeur : la valeur LACLE ne désignant pas un plan de classement, le prédicat (*jtype* = LACLE) rend nécessairement l'ensemble vide $\emptyset$.

L'attribut *irole* contient le rôle d'un sujet de l'acte.

Cette relation de gestion fournit un instrument pour classer les actes selon des critères multiples, à partir de clés définies ou calculées, et améliorer l'efficacité des calculs.

## 3.10 Variables hors classes

La valeur NBCLASSE de la variable *jclasse* indique que les valeurs textuelles associées n'appartiennent pas au langage $H$. Celà permet d'archiver du texte libre recopié tel quel dans l'attribut *segment* de la relation

**Libre(isource,iacte, irole, jtype,k,***rupture,segment***).**

tout en conservant le lien avec les autres variables définies sur $\mathcal{H}$.

# Annexe A

# Conventions PAGI

Les conventions actuellement utilisées à l'INRA pour les données de l'état civil français sont décrites dans cette annexe.

L'attribut *jfpagi* du pilote contient la valeur de la variable *fpagi* de gestion du format PAGI :

*fpagi* = 0 traduit une indétermination ou l'unicité du format dans la classe ;

*fpagi* =NBFPAGI signale que le composant PAGI code un évènement, quelle que soit la classe ;

*fpagi* $\in$ {1...NBFPAGI−1} désigne un format PAGI ou identifie un plan de classement.

Les plans de classement sont utilisés pour construire les relations d'équivalence des dictionnaires et classer les actes en déclarant des variables dans la classe GESTION.

Les valeurs prédéfinies des variables utilisées dans le calcul de la variable *type* du paragraphe 1.4.2 sont transmises dans l'attribut *jtype* du pilote.

Dans les notations qui suivent, les formes minuscule et majuscule d'une lettre sont équivalentes, les conversions éventuelles étant faites dans les fonctions pour prodnire les graphies normalisées.

## A.1 Codage PAGI des évènements

La valeur de la variable *evenement* est lue dans l'acte pour les mentions marginales et les dates. Cette reconnaissance dynamique est également pos-

sible pour les lieux mais n'est pas utilisée actuellement. L'évènement est codé dans le composant PAGI de format NBFPAGI au moyen d'une lettre :

**N** : naissance,
**M** : mariage,
**D** : décès,
**S** : divorce,
**J** : jugement,
**L** : liste de personnes,
**T** : transcription,
**H** : succession.

### Rang des mentions marginales

Le rang des mentions marginales est géré dans la variable *sequence* définie au paragraphe 1.4.2. Dans l'univers d'acquisition PAGI, des champs consécutifs sont réservés pour un nombre fixe de mentions marginales. La valeur de la variable *sequence* est prédéfinie et doit en conséquence être affectée dans l'attribut *jtype* du pilote pour les composants PAGI concernés.

## A.2 Classe PERSONNE

### Formats PAGI

**1** : nom ;
**2** : liste des prénoms ;
**3** : nom , liste des prenoms ;
**4** : prénom ;
**5** : sexe ;
**6** : legitimité ;
**7** : reconnaissance d'enfant ;
**8** : etat matrimonial;
**9** ... NBFPAGI-**1** : plans de classement.

Les expressions parenthèsées sont admises pour les trois premiers formats PAGI. En l'absence de préfixe, un élément de liste est par défaut un surnom.

### Autres noms

Les autres noms sont introduits par des préfixes dérivés de $< gtag >$ :

**d** : dit ou dite (surnom),

**e, é**: épouse ou époux,
**v**: veuve ou veuf,
**n, né**: née.

### Notes et indications

Les notes et indications sont dérivées de $< htag >$ :

**i**: notes et indications.

### Sexe

H : homme,
F : femme.

### Légitimité d'un enfant

**L** : enfant légitime,
**N** : enfant naturel,
**T** : enfant trouvé de parents inconnus,
**A** : information absente.

### Reconnaissance d'un enfant

**H** : reconnu par le père,
**F** : reconnu par la mère,
**P** : reconnu par les parents.

### État matrimonial

**C** : célibataire,
**M** : marié,
**V** : veuf,
**S** : divorcé,
**A** : information absente.

## A.3 Classe PERIODE

### Formats PAGI

**1** : date complète ;

**2** : jour ;
**3** : mois ;
**4** : année ;
**5** : âge ;
**6** : délai de déclaration.

### Unités de mesure de l'âge

**a** : années,
**m** : mois,
**s** : semaines,
**j** : jours,
**h** : heures.

## A.4 Classe LIEU

L'environnement PAGI utilise une structure syntaxique propre afin de préfixer les toponymes par les codes des communes et départements, et parfois des anciennes paroisses : « *35188A-Montfort-sur-Meu(Coulon)* » représente l'ancienne paroisse de Coulon (*A*) de la commune de Montfort-sur-Meu (188) en Ille-et-Vilaine (35). Cette forme est dite lieu PAGI.

### Lieux PAGI

Les lieux PAGI sont des valeurs alphanumériques générées par les les règles suivantes :

$< lieupagi > \longrightarrow < codepagi > [- < toponyme >]$
$< codepagi > \longrightarrow < dept >< commune > [< paroisse >]$
$< dept > \longrightarrow < arabe >< arabe >$
$< commune > \longrightarrow < arabe >< arabe >< arabe >$
$< paroisse > \longrightarrow < latin > [< latin >]$

Le toponyme est nécessairement préfixé par un code de cinq chiffres pour coder une commune d'un département : la ville de Nice sera codée « *06088-Nice* » et non « *6088-Nice* ». Ce code est suivi optionnellement d'une ou deux lettres pour identifier une ancienne paroisse. Dans ce dernier cas, le nom de la paroisse est placé entre parenthèses à droite.

Le passage par cette forme pour l'acquisition de nouvelles données n'est plus nécessaire : il suffit de se donner les variables

$$commune, departement \in \mathcal{N} \text{ et } lieu \in \mathcal{H}$$

qui pour l'exemple cité recevront respectivement les valeurs

188, 35 et « *Montfort-sur-Meu(p:Coulon)* ».

Le code de la paroisse sera recherché dans un dictionnaire.

### Format PAGI

**1** : forme syntaxique générale $< cheflieu > [([< tag >:] < groupe >)]$ ;
**2** : code département ;
**3** : code commune ;
**4** : code section ;
**5** : département en clair ;
**6** : chef-lieu et toponyme ;
**7** : chef-lieu et adresse ;
**8** : chef-lieu et note ;
**9** : lieu PAGI ;
**10** : code graphie ;
**11** ... NBFPAGI-**1** : plans de classement.[1]

Les expressions parenthèsées sont autorisées pour les formats PAGI 1, 6, 7, 8, 9. En l'absence de préfixe, un élément de liste est par défaut un hameau.

### Subdivisions

Les subdivisions sont des toponymes dérivés de $< gtag >$ :

**h** : hameau,
**p** : paroisse,
**c** : commune,
**v** : groupe de hameaux dit village,
**q** : quartier,
**e**, **é** : écart,
**x** : non identifié.

[1] Le code postal parfois utilisé est vu comme un plan de classement.

### Adresses et indications

Les adresses et indications sont dérivées de $< htag >$ :

**a** : adresse,
**i** : notes et indications.

## A.5 Classe STATUT

### Format PAGI

**1** : signature ;
**2** : liste de métiers et activités ;
**3** ... NBFPAGI-**1** : plans de classement.

### Signature

**S** : a signé,
**N** : a déclaré ne pas savoir signer,
**A** : personne absente.

### Métiers et activités

Les métiers et activités sont saisis dans des listes dont les éléments sont les mots du langage $H$, avec éventuellement des expressions parenthèsées.

## A.6 Classe GESTION

La valeur LACLE du format PAGI désigne le composant qui contient la valeur de la clé d'identification.

Les autres valeurs sont des numéros de plans de classement associés à des composants PAGI généralement déclarés simultanément dans d'autres classes comme clé numérique d'identification.

Les formats PAGI suivants permettent de lire une clé d'identification et de classer les actes de mariage selon les lieux d'origine des époux.

1 : clé d'identification (LACLE= 1,
2 : département d'origine du marié ;
3 : commune d'origine du marié ;
4 : département d'origine de la femme ;
5 : commune d'origine de la femme ;

# Table des matières

Un document historique élémentaire décrit un évènement survenu en un jour et en un lieu, lequel met en scène des personnes avec des rôles bien établis.

L'enquête réalisée conjointement avec le CNRS et l'INED sur la mobilité géographique et sociale des descendants de trois mille familles mariées au début du $XVIII^{ème}$ siècle regroupe des centaines de milliers de ces documents historiques.

Le langage *H* a été conçu pour formaliser et décrire dans un langage mathématique simple ce type de données factuelles, avec pour valeurs des dates, des noms, des lieux et des métiers. Il est décrit par une grammaire formelle et une traduction a été faite pour les besoins de la dite enquête.

Ce rapport présente ce langage formel et l'ensemble des conventions adoptées pour modéliser la représentation des données extraites des documents d'archives. Le compilateur réalisé permet de normaliser la représentation des valeurs à partir des données brutes pour les affecter ensuite dans des relations gérées dans une base de données.

HISTOIRE
INFORMATIQUE
LANGAGES FORMELS