



IV - Identifiants

OBJECTIF : L'identifiant d'un objet est d'une importance capitale dès lors que l'on souhaite gérer une base de données car sans lui, il est impossible de distinguer un enregistrement (ou un objet) de façon unique dans une table. De plus, un identifiant unique facilite les recherches, évite les doublons, permet les jointures et relations (cf. fiches « Sélections et jointures »). Cette fiche explique la construction et la gestion de cet identifiant unique en insistant notamment sur la notion de clé primaire à stabiliser et sur le risque d'utiliser l'identifiant automatique généré par certains logiciels.

IV.1 - PREREQUIS

IV.1.1 - Structuration des données

Comme vu au travers du guide méthodologique (chapitre II.3.4) et des différentes fiches techniques l'identifiant est essentiel à la gestion des données. Partie intégrante du fonctionnement des bases de données relationnelles, il est intimement lié à la structuration des données et doit être prévu en amont de la création des données (cf. fiche « Structuration des données »).

IV.2 - NOTE METHODOLOGIQUE

IV.2.1 - Différents champs contenant des identifiants

Une même table peut comporter plusieurs champs contenant des identifiants. Par exemple la couche "Bocage_Ajour_L" comporte les champs "ID_AJOUR", "ID_EDL" et "ID_TRVX". Mais attention, ces différents champs n'ont pas la même vocation (cf. figure 1).

Clé primaire

Dans une base de données relationnelle, une **clé primaire** est une contrainte d'unicité qui permet d'identifier de manière unique un enregistrement dans une table. Ainsi, dans la couche "Bocage_Ajour_L", le champ "ID_AJOUR" est une clé primaire. **Bien souvent, la clé primaire s'appuie sur un seul champ, mais elle peut aussi être composée de plusieurs champs de la table.** Une clé primaire doit répondre aux exigences suivantes :

- deux enregistrements (ou objets) distincts d'une table ne peuvent pas avoir les mêmes valeurs pour les champs définis au niveau de la clé primaire.
- la clé primaire ne doit pas évoluer dans le temps, elle est donnée une fois pour tout à l'objet.

Exemple : le champ "ID_EDL" est celui contenant la clé primaire de la table "Bocage_EdL_XXXX_L".

Clés étrangères

Dans une table, d'autres champs peuvent contenir des identifiants. Ce sont alors des **clés étrangères** faisant référence à la clé primaire d'une autre table. Le fait de faire correspondre une clef externe à une clef primaire met en place une relation entre tables.

Exemple : dans la couche "Bocage_AJour_L" le champ "ID_AJOUR" contient la clef primaire, tandis que le champ "ID_EDL" contient la clef étrangère permettant de mettre en relation la table "Bocage_EdL_XXXX_L" avec la table "Bocage_AJour_L" en faisant correspondre les "ID_EDL". Cependant, la couche "Bocage_AJour_L" contient également des objets issus des couches travaux, aussi la clé étrangère "ID_TRVX" permet de la mettre en relation avec la couche "Bocage_Trvx_XXXY_L" grâce à sa clé primaire "ID_TRVX".

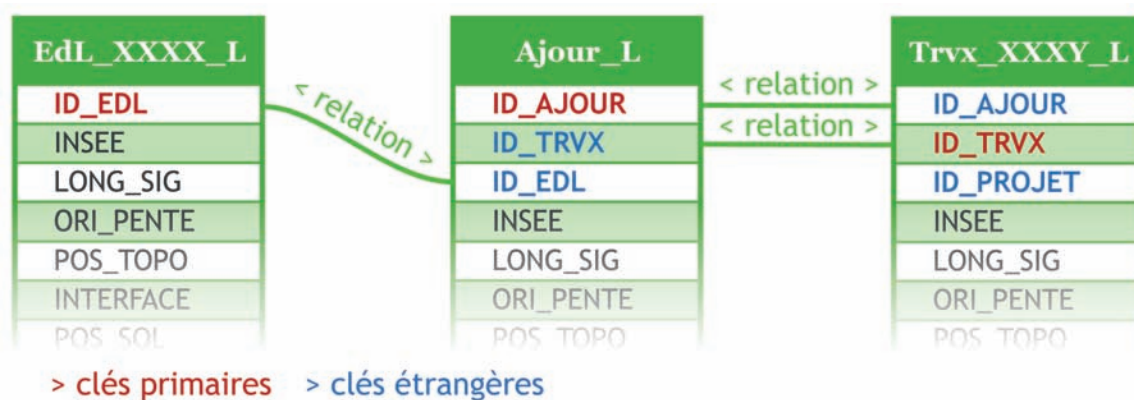


Figure 1 - Clé primaire, clés étrangères et relations

Dans cette fiche nous allons nous intéresser à la constitution de la clé primaire.

Clé primaire d'une table et identifiant unique d'un objet

Il existe donc une clé primaire pour chaque table. La gestion d'un S.I.G. opérationnel en local implique qu'un même objet haie (sur le terrain) puisse être suivi au travers de son identifiant primaire dans toutes les tables en production dans lesquelles il peut se retrouver (à jour, travaux...). Cette gymnastique, qui peut paraître compliquée de prime abord, est explicitée par un exemple concret ci-dessous.

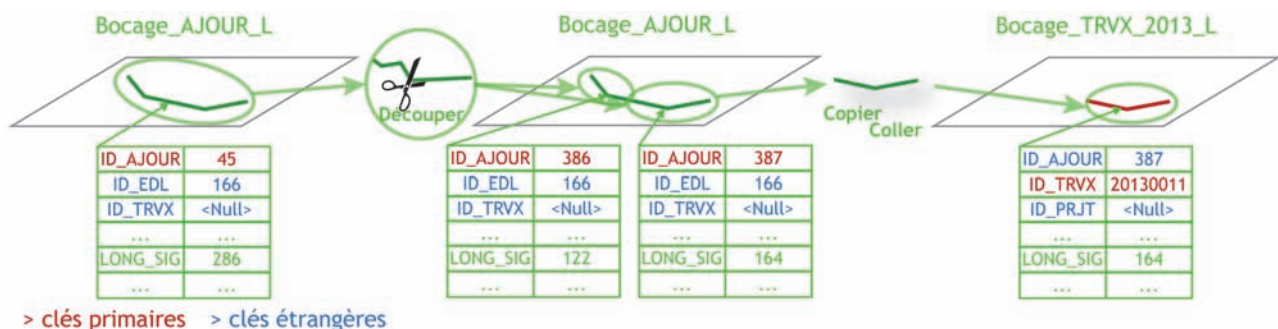


Figure 2 - Exemple du suivi d'identifiant dans les différentes couche du S.I.G bocage local

Explication du schéma : Il est prévu de regarnir une partie d'un linéaire bocager vieillissant. Ce linéaire est présent dans le dernier état des lieux bocager dans la couche "Bocage_EDL_2008_L" avec l'"ID_EDL=166". Si les préconisations de structuration ont bien été suivies, l'état des lieux est repris par

copie et constitue le socle de la table “Bocage_AJour_L”. Avant travaux, ce linéaire est présent dans la couche “Bocage_AJour_L” avec l’“ID_EDL=166” et une clé primaire pour cette couche : “ID_AJOUR=45”.

Lorsque des travaux d’entretien sont réalisés, cette modification doit être intégrée dans la base Bocage. C’est dans la couche “Bocage_AJour_L” que l’on fait l’opération en coupant le linéaire en deux parties. Deux nouveaux identifiants (exemple : “ID_AJOUR=386” et “ID_AJOUR=387”) mais, **attention : il est également nécessaire de modifier les longueurs des deux nouveaux objets (avec la fonction longueur) ou tout autre champ modifié par l’opération.**

On peut désormais copier l’objet identifié “ID_AJOUR=387” et le coller dans la couche travaux de l’année “Bocage_Trvx_2013_L”. Dans la couche travaux, on crée alors un identifiant travaux “ID_TRVX=2013011”, qui devient la clé primaire de cette couche Travaux.

Il est fondamental de noter qu’il existe une clé primaire pour chaque table. Mais aussi qu’un identifiant unique permet de suivre l’objet dans toutes les couches en production. C’est l’“ID_AJOUR=387” de la couche “Bocage_AJour_L” qui doit être retenu comme identifiant unique car chaque tronçon qui va avoir une réalité sur le terrain (création / travaux) se retrouve à chaque fin de campagne dans la couche “Bocage_AJour_L”.

3

Attention : Les couches “Bocage_EdL_2008_L” et “Bocage_Projet_L” ne sont pas des couches en production. La première est une donnée référentielle, la seconde rassemble des éléments qui n’ont pas encore de réalité de terrain, l’identifiant unique n’est pas visualisable sur ces exceptions.

Pour mieux comprendre le pourquoi de ces manipulations reportez-vous aux fiches « Structuration des données » & « Gestion du S.I.G bocage »). La suite de cette fiche explique la façon dont on construit les différents types d’identifiants évoqués ci-dessus.

IV.2.2 - Différentes formes d’identifiants

La forme de l’identifiant d’objet, clé primaire, est libre tant qu’elle permet de créer des références uniques. Un identifiant peut donc être du type nombre (se sera alors forcément un entier long car les identifiants ne connaissent a priori pas de limite), chaîne de caractères voire date (clé primaire d’un calendrier).

Numéros automatiques et compteurs

Dans sa forme la plus simple un identifiant peut être un numéro automatique équivalent à un simple compteur de ligne. Le premier enregistrement aura l’identifiant “1”, le deuxième “2”, le cent quinzième “115”, etc.

Ce numéro automatique est l’identifiant par défaut donné par les logiciels de SIG (ou de gestion de base de données). On retrouvera ainsi les “FID”, le “OID” ou le “ROWID” (voir § IV.3.1) mais ils sont à utiliser avec prudence (cf. IV.2.3).

Identifiants composés

Pour éviter les doublons d'identifiants d'objets, notamment entre différentes tables historisées (par exemple les couches "Bocages_Trvx_XXXX") ou tout simplement pour avoir un identifiant signifiant **il faut composer la clé primaire**. Pour ce faire on pratiquera une concaténation¹ de plusieurs champs, d'un champ et d'un compteur ou d'un terme fixe (numérique ou caractères) et d'un compteur. **Le but final étant toujours que l'identifiant soit unique.**

Exemple : pour la clé primaire de la couche "Bocage_Trvx_XXXX_L" on pratiquera la concaténation de l'année "XXXX" et d'un compteur, ce qui donnera par exemple : XXXY00023 (voir § IV.3.2).

Import de données dans une base de donnée plus large

Imaginons maintenant que vous transmettiez vos données bocage à jour pour qu'elles soient intégrées à un référentiel bocage régional. Si vous, comme les autres, avez suivi les préconisations données, alors il y a de fortes chances pour que dans la couche "Bocage_AJOUR_L" du référentiel régional il y ait plusieurs linéaires bocagers dont la clé primaire soit par exemple "237". La clé primaire, pourtant unique chez vous, n'est plus unique dans le référentiel régional. Pour y palier il faudra définir une clé primaire composé de deux champs. On ajoutera alors un champ "SIRET" destiné à contenir le numéro SIRET de la structure productrice. En effet, le numéro SIRET est lui même une clé primaire, identifiant unique de structure. Si le numéro SIRET de votre structure est le "50755093700019" alors le champ "SIRET" sera partout égal à ce code. Au niveau régional on composera une clé primaire faite des deux champs "ID_AJOUR" et "SIRET", et on identifiera alors a coup sûr un objet haie unique.

IV.2.3 - Stabiliser la clé primaire

L'un des principes primordiaux de la clé primaire est qu'elle soit stable dans le temps quelles que soient les manipulations futures opérées sur les données. Or, les numéros automatiques par défaut de type "FID", "OID" ou "ROWID" ne résistent pas à toutes les manipulations (voir § IV.3.1). Il faut donc stabiliser la clé primaire en l'enregistrant dans un champ spécifique dédié, et **ne plus jamais y toucher une fois l'attribution faite** (voir § IV.3.1). C'est le rôle des champs "ID_EDL", "ID_AJOUR" et "ID_TRVX" respectivement pour les tables "Bocage_EdL", "Bocage_Ajour" et "Bocage_Trvx".

IV.2.4 - Petit point sur les formats de fichiers de données géographiques vectorielles

Exceptionnellement pour cette fiche technique, les manipulations étant très liées au logiciel de SIG utilisé, ou plutôt au format de fichier utilisé (*Shape*, *Tab* ou *GeoDataBase*) nous allons ici détailler les manipulations spécifiques.

Fichiers *shape* « .shp »

Le format *shape* « .shp » est le format d'origine d'ArcView®. Etant passé dans le domaine public depuis de nombreuses années il est désormais largement utilisé et notamment par les logiciels libres (gvSIG, QGIS...).

¹ Le terme concaténation, désigne l'action de mettre bout à bout au moins deux chaînes.

GeoDataBases personnelles et fichiers « .mdb » et « .gdb »

Les *GeoDataBases* (pour Base de Données Géographique) est, depuis l'avènement d'ArcGIS® 8, le format de prédilection sous ArcGIS®. Il existe deux types de *GeoDataBases* :

- Les *GeoDataBases* **personnelles**, basées sur le format « .mdb » de Microsoft Access®, leur utilisation est désormais déconseillée par ESRI pour des problèmes de stabilité. Ce format est passé très récemment dans le domaine public et sera donc peut-être supporté prochainement par d'autres logiciels.
- Les *GeoDataBases* **fichiers**, basées sur un système de fichiers, existent depuis ArcGIS® 9.x, c'est un format propriétaire ESRI qui n'est donc utilisable que sous ArcGIS®.

Cependant, pour l'utilisateur simple, le fait qu'une *GeoDataBase* soit de type personnelle ou fichiers n'a que peu d'incidences.

Fichiers Tab

Le format Tab « .tab » est le format propriétaire de MapInfo®. QGIS supporte également le format TAB, mais son utilisation reste largement cantonnée à MapInfo®.

IV.3 - MANIPULATIONS

IV.3.1 - Récupération du numéro automatique

Dans un fichier *shape* « .shp »

Les fichiers de formes *shape* travaillent avec un identifiant donné par défaut le "FID" (pour *Feature Identifier* - identifiant d'entité). Celui-ci est contenu dans le champ portant le même nom qui n'est pas modifiable car totalement géré par le logiciel de SIG. Mais **attention cet identifiant "FID" n'est pas une clef primaire fiable**. En effet, comme le montre la figure ci-dessous il évolue au cours de la suppression d'objets.

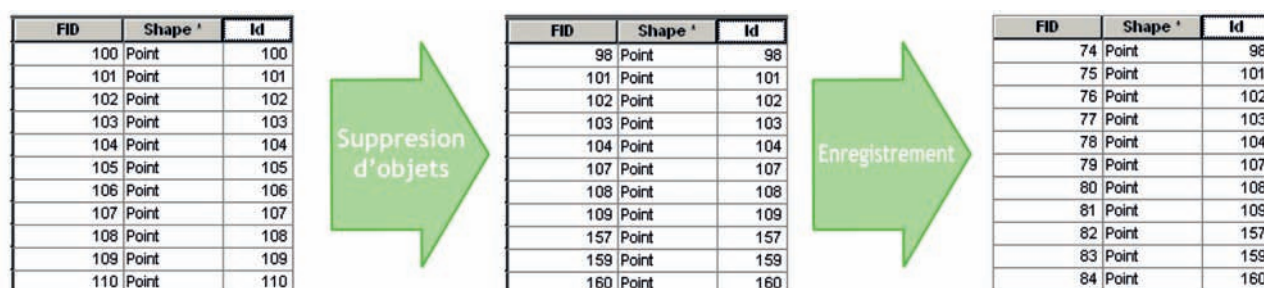


Figure 3 - Evolution du FID lors de la suppression d'objet

Il convient donc de récupérer ce FID pour l'enregistrer seul (champ ID dans l'exemple ci-dessus) ou concaténé à d'autres valeurs dans un champ dédié qui servira de clé primaire.

➤ ArcGIS C05, QGIS C06, gvSIG C06

Le "FID" des fichiers de formes *shape* est masqué sous QGIS et gvSIG. Pour l'obtenir, sous QGIS il faudra passer par "\$id" des fonctions « Enregistrement » et sous gvSIG par la fonction "rec()" :

- QGIS : **\$id**
- gvSIG : **rec()**
- ArcGIS® 9.x : **[FID]**

ATTENTION : Le « FID » n'étant qu'un compteur de ligne il n'est pas stable (voir figure ci-dessus). Aussi, une attention toute particulière devra être portée lors de l'attribution des identifiants (clé primaire) aux objets. Si une partie des objets a déjà un identifiant, il ne faut surtout pas le modifier ! Les manipulations ne doivent donc être faites que sur les objets n'ayant pas encore d'identifiant (sélection préalable et calculs uniquement sur objets sélectionnés). D'autre part, pour être sûr qu'il n'y ait pas de doublon, il faudra ajouter un nombre au « FID » (en effet, le « FID » d'un nouvel enregistrement peut être inférieur au dernier identifiant attribué).

Dans une GeoDatabase personnelle ou fichiers « .mdb » et « .gdb »

A l'image du « FID » du fichier *shape*, l'« OBJECTID » ou « OID » (pour *Object Identifier* - identifiant d'objet) contenu dans le champ « OBJECTID » des *GeoDataBases* est un numéro automatique géré par ArcGIS®. Cependant, celui-ci est beaucoup plus fiable que le « FID ». En effet, tant que l'on reste dans la même table il est stable. Si on supprime, par exemple, les enregistrements « 101 » à « 105 » d'une table donnée, ces identifiants ne seront plus jamais utilisés. Il est bien unique et c'est pourquoi, au fil des suppressions et ajouts, on peut se retrouver avec des valeurs d'« OBJECTID » qui ne se suivent pas (voir figure 3).

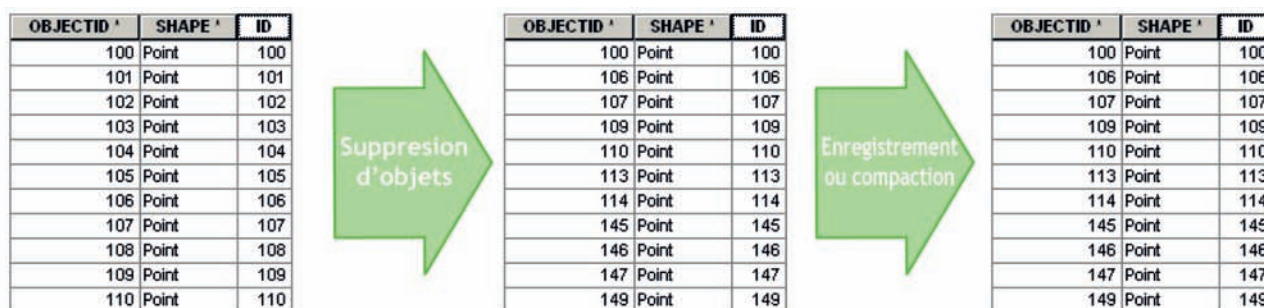


Figure 4 - Stabilité de l'OBJECTID malgré les suppressions

Par contre, si on l'exporte une partie des données (et donc à fortiori si on importe des données), l'objet ayant par exemple l'« OBJECTID » « 146 » aura dans la table exportée un nouveau « FID », par exemple « 9 » (voir figure 4).

Le « FID » peut donc être utilisé comme base pour la constitution d'une clé primaire mais ne doit pas être utilisé tel quel.



Figure 5 - Evolution de l'OID lors de l'export de données

Tout comme pour les fichiers *shape* il conviendra donc de récupérer ce numéro automatique pour l'enregistrer seul ou concaténé à d'autres valeurs dans un champ dédié qui servira de clé primaire (champ « ID » dans les figures ci-dessus).

> ArcGIS C05

Dans un fichier Tab

Dans le format *TAB* « .tab » aucun champ identifiant clé primaire n'apparaît par défaut dans les tables. Pourtant MapInfo® pour son propre fonctionnement a besoin de mettre en relation table de données et objets géométrique. Il se base donc sur une clé primaire "ROWID" (pour *Row Identifier* – Identifiant de ligne) qui, comme son nom l'indique, est de type compteur de ligne, mais le "ROWID" n'apparaît pas dans un champ de la table. Il faudra donc récupérer ce "ROWID" pour l'enregistrer seul ou concaténé à d'autres valeurs dans un champ dédié qui servira de clé primaire. Pour se faire on utilisera tout simplement l'instruction "ROWID" lors de la mise à jour de colonne.

> MapInfo C05



Figure 6 - Evolution du ROWID lors de la suppression d'objet

ATTENTION : Tout comme le "FID", le "ROWID" n'est qu'un compteur de ligne, il n'est donc pas stable. Les mêmes précautions qu'à l'utilisation des fichiers de *shape* sont donc à prendre avec les fichiers TAB.

IV.3.2 - Concaténation

Comme évoqué dans le paragraphe « Identifiants composés », il s'avère souvent nécessaire de créer des identifiants (clé primaire) signifiants par concaténation :

- de plusieurs champs de la table,
- d'un champ de la table et d'un compteur ("FID", "ROWID", "OID")
- d'un terme (nombre ou chaîne de caractères) et d'un compteur

Pour cela on va procéder à des opérations dans l'interface de « mise à jour de colonne » ou de « calcul de champs » (différents termes selon les logiciels). Ces opérations devront être écrites dans le langage de calcul du logiciel qui peut être SQL, VBA² (ArcGIS® 8.x à 10.0), Python² (ArcGIS® 10.x) ou plus souvent un langage propre au logiciel se rapprochant d'un des langages précédents (par exemple Jython – variante de Python – pour gvSIG).

> ArcGIS C05, MapInfo C05, QGIS C06, gvSIG C06

² VBA et Python sont des langages de programmation permettant d'écrire des petits script ou des logiciels complets. VBA pour Visual Basic for Application est un langage Microsoft très utilisé dans les logiciels de la marque mais de moins en moins utilisé ailleurs et d'ailleurs non compatible avec ArcGIS® depuis la version 10.1. Python quant à lui est le langage géomatique qui monte (implémenté dans ArcGIS® depuis la version 9.3). Open Source ce langage est donc largement utilisé dans les logiciels SIG open source (certains sont d'ailleurs complètement développés en Python).

Concaténation de chaînes de caractères

Le principe de la concaténation est de **créer une nouvelle chaîne de caractère en mettant bout à bout au moins deux chaînes de caractères**. Par exemple la concaténation de « bo » et de « cage » donne « bo-cage ». Selon le logiciel utilisé on réalisera une concaténation de la manière suivante :

- QGIS : `'CHAINE_1' || 'CHAINE_2'`
- gvSIG : `"CHAINE_1" + "CHAINE_2"`
- ArcGIS® 9.x - VBA : `"CHAINE_1" & "CHAINE_2"`
- ArcGIS® 10.x - Python : `"CHAINE_1" + "CHAINE_2"`
- MapInfo® : `"CHAINE_1" + "CHAINE_2"`

Convertir un nombre en chaîne de caractères

Souvent on voudra concaténer des nombres entre eux ou des nombres avec des chaînes de caractères pour obtenir, au final, une chaîne de caractères. Or, nombre et chaîne de caractères sont deux types différents pour un système de gestion de base de données. Pour pouvoir concaténer un nombre à une chaîne de caractère il va donc falloir le convertir en chaîne de caractères. Selon les langages on réalisera une concaténation d'une chaîne de caractère et d'un nombre de la manière suivante :

- QGIS : `'CHAINE_1' || toString(NOMBRE)`
- gvSIG : `"CHAINE_1" + toString(NOMBRE)`
- ArcGIS® 9.x - VBA : `"CHAINE_1" & CStr(NOMBRE)`
- ArcGIS® 10.x - Python : `"CHAINE_1" + toString(NOMBRE)`
- MapInfo® : `"CHAINE_1" + NOMBRE`

Concaténation de nombres

Bien que, par définition, la concaténation soit le fait de mettre bout à bout au moins deux chaînes de caractères, on peut aussi parler de concaténation de nombres. Attention, **nous parlons bien ici d'une concaténation d'au moins deux nombres dont le résultat est un nombre**.

En effet, souvent on souhaitera concaténer un préfixe (de type entier) avec un compteur (de type entier également), tout ceci en gardant le même nombre de caractère dans la chaîne résultante quelque soit le nombre du compteur. Autrement dit, on ne voudra pas « 20085 », « 200837 », « 2008328 »...etc mais « 20080005 », « 20080035 », « 20080328 »...etc.

Pour cela on va jouer un peu sur les mots, car on ne concatène pas les chiffres, mais on réalise des opérations dessus (addition, soustraction, multiplication, division...). Pour obtenir un identifiant comportant un préfixe nombre (année par exemple) et 4 caractères réservés au compteur on procède de la manière suivante :

- QGIS : $(ANNEE * 10000) + "ID_TRVX"$
- gvSIG : $(ANNEE * 10000) + [ID_TRVX]$
- ArcGIS® 9.x – VBA – fichier *shape* : $(ANNEE * 10000) + [ID_TRVX]$
- ArcGIS® 10.x – Python – GDB fichiers : $(ANNEE * 10000) + "ID_TRVX"$
- MapInfo® : $(ANNEE * 10000) + ID_TRVX$

Par exemple : « 2008 * 10000 » donne « 20080000 » auquel on ajoute le compteur par exemple « 327 » soit « 20080327 ».

```
> ArcGIS_C05, MapInfo_C05, QGIS_C06, gvSIG_C06
```

ATTENTION : Ce type d'identifiant composé oblige l'utilisateur à prévoir, dès le début, le nombre maximum d'objets. En effet, si l'on multiplie le préfixe par 10 cela laisse la possibilité d'identifier seulement 10 objets (de 0 à 10), si l'on multiplie par 100, 100 objets et ainsi de suite.

IV.4 - RESUME - QUELLE PRECONISATION POUR UN SIG BOCAGE ?

Pour rappel, il est fondamental de noter que :

- Dans un S.I.G local, l'identifiant unique des couches en production est celui des couches "Bocage_AJour_L" et "Bocage_AJour_S" qui est intitulé "ID_AJOUR" et est construit comme précédemment décrit (cf. figure 1).
- Seules les couches "Bocage_EdL_XXXX_*" et "Bocage_Projet_*" ne possèdent pas cet identifiant unique du fait de leurs conditions de création (cf. § IV.2.1 « Clé primaire d'une table et identifiant unique d'un objet »).
- Lorsqu'une couche sort de la structure (pour un échange de données avec d'autres partenaires ou pour transmettre des données au niveau régional), l'ajout d'un champ SIRET permet de créer un identifiant unique au niveau propre à la structure.

La clé primaire est alors constituée de deux champs (cf § IV.2.1 « Clé primaire »).



Cette œuvre est mise à disposition selon les termes de la [Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions 3.0 non transposé](#)