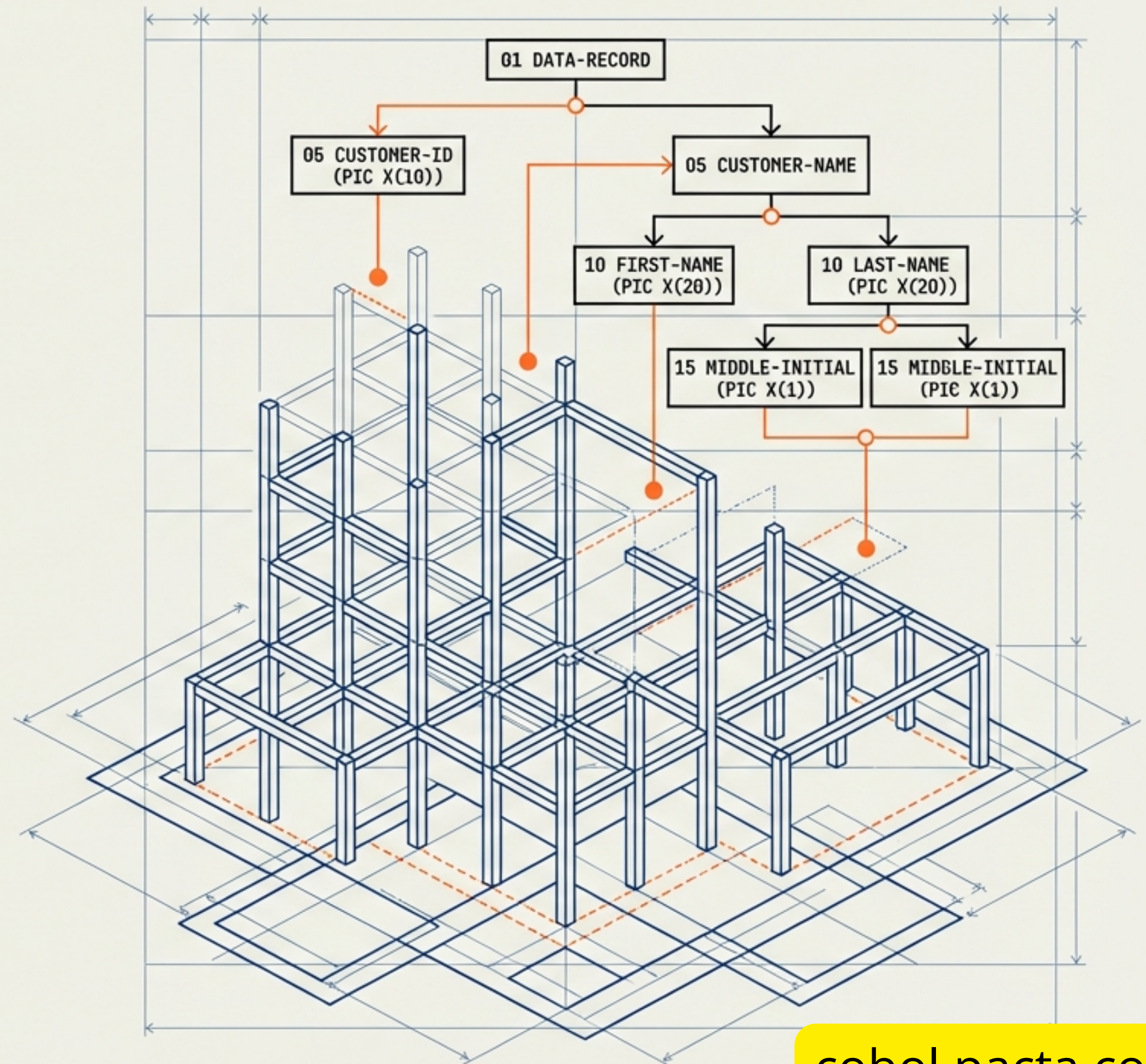


Maîtriser les Niveaux en COBOL

L'art de structurer et modéliser les données comme un architecte.

Les déclarations de données en COBOL sont bien plus que des variables ; elles représentent une hiérarchie architecturale complète. Ce document est un manuel pour construire, organiser et affiner cette structure.

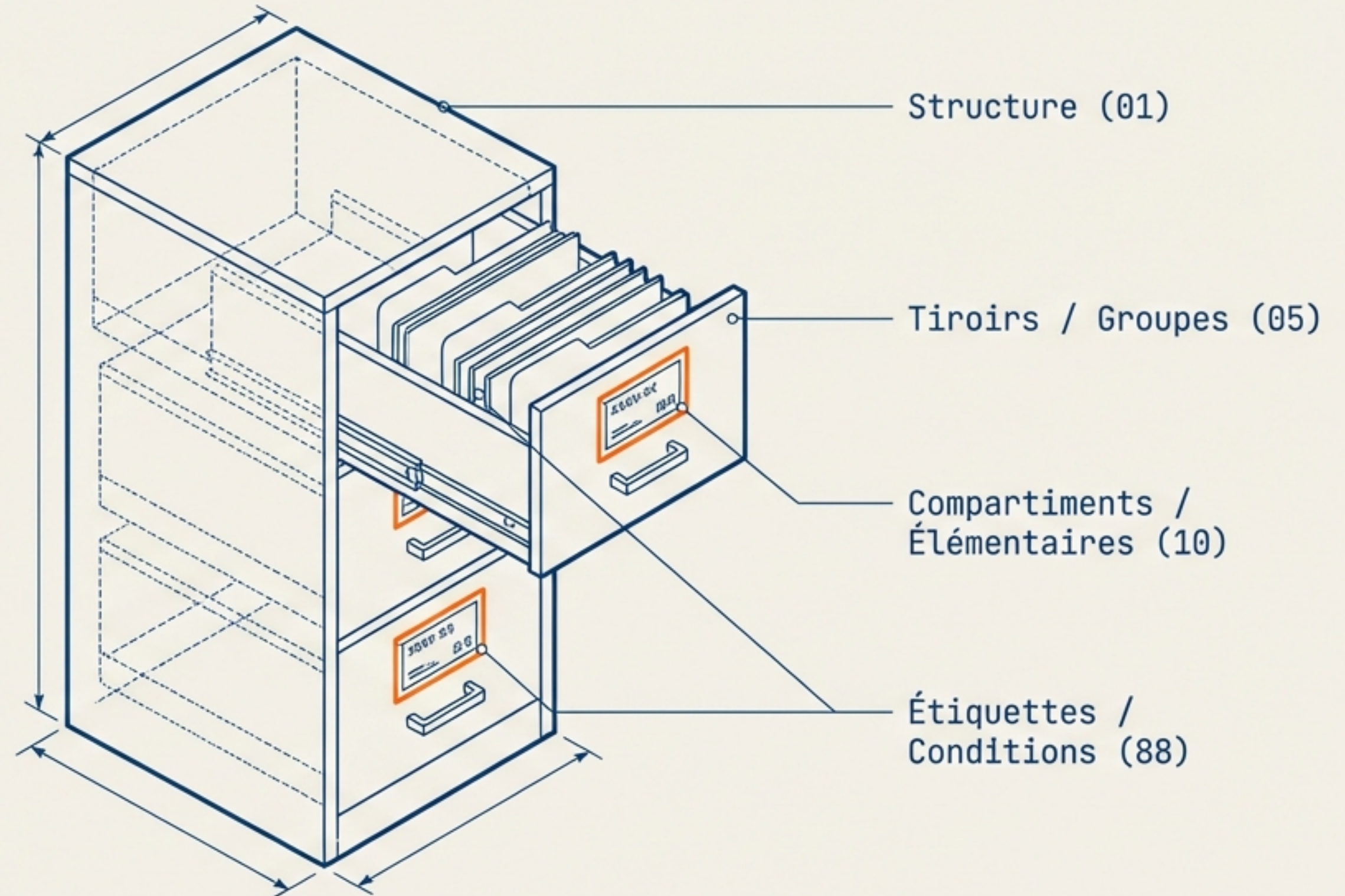


La Philosophie : Plus que des Variables

Déclarer des données, c'est construire une armoire.

Ce mécanisme donne au COBOL une force unique : la capacité de manipuler des champs précis (ex: code postal) et des blocs et des blocs massifs (ex: dossier client) simultanément.

Le "Numéro de niveau" indique la position exacte dans cette hiérarchie.



Vue d'Ensemble : Deux Familles de Niveaux

Les Niveaux Généraux

01 – 49

- Fonction : Établir la hiérarchie
- 01 : La Racine (Enveloppe)
- 02-49 : Les Branches (Sous-éléments)



Les Niveaux Spéciaux

66 77 88

- 66 : Vue Alternative (RENAMES)
- 77 : Variable Isolée (Indépendant)
- 88 : Condition Sémantique (Noms de condition)



L'Arborescence 01 à 49

01 FICHE-ABONNE.

05 ID-ABONNE

PIC 9(6).

05 NOM-COMPLET.

10 NOM-FAMILLE PIC X(18).

10 PRENOM PIC X(12).

05 ADRESSE-POSTALE.

FICHE-ABONNE
(01)

ID-ABONNE (05)

NOM-COMPLET (05)

NOM-FAMILLE (10)

PRENOM (10)

ADRESSE-POSTALE (05)

La logique descend :
10 dépend de
05, qui dépend
de 01.

Groupe vs Élémentaire

Le Groupe (Conteneur)



- Un assemblage de plusieurs sous-champs.
- Règle : Pas de clause PIC.
- Exemple : **05 NOM-COMPLET.**

L'Élémentaire (Donnée Finale)



- Le champ final contenant la donnée réelle.
- Règle : Doit avoir un PIC (taille/type).
- Exemple : **10 NOM-FAMILLE PIC X(18).**

Pourquoi est-ce important ?

Manipulation : On peut déplacer tout le groupe (MOVE FICHE-ABONNE) ou juste un champ (MOVE 'LYON').

Affichage : DISPLAY NOM-COMPLET montre la concaténation brute (Nom + Prénom).

La Logique de Numérotation

Règle d'Or : On descend dans le détail.



Structure Illogique

```
01 BLOC
  10 CHAMP-A
    05 GROUPE
```



Interdit de 'remonter' de 10 à 05 dans le même contexte.



Structure Valide

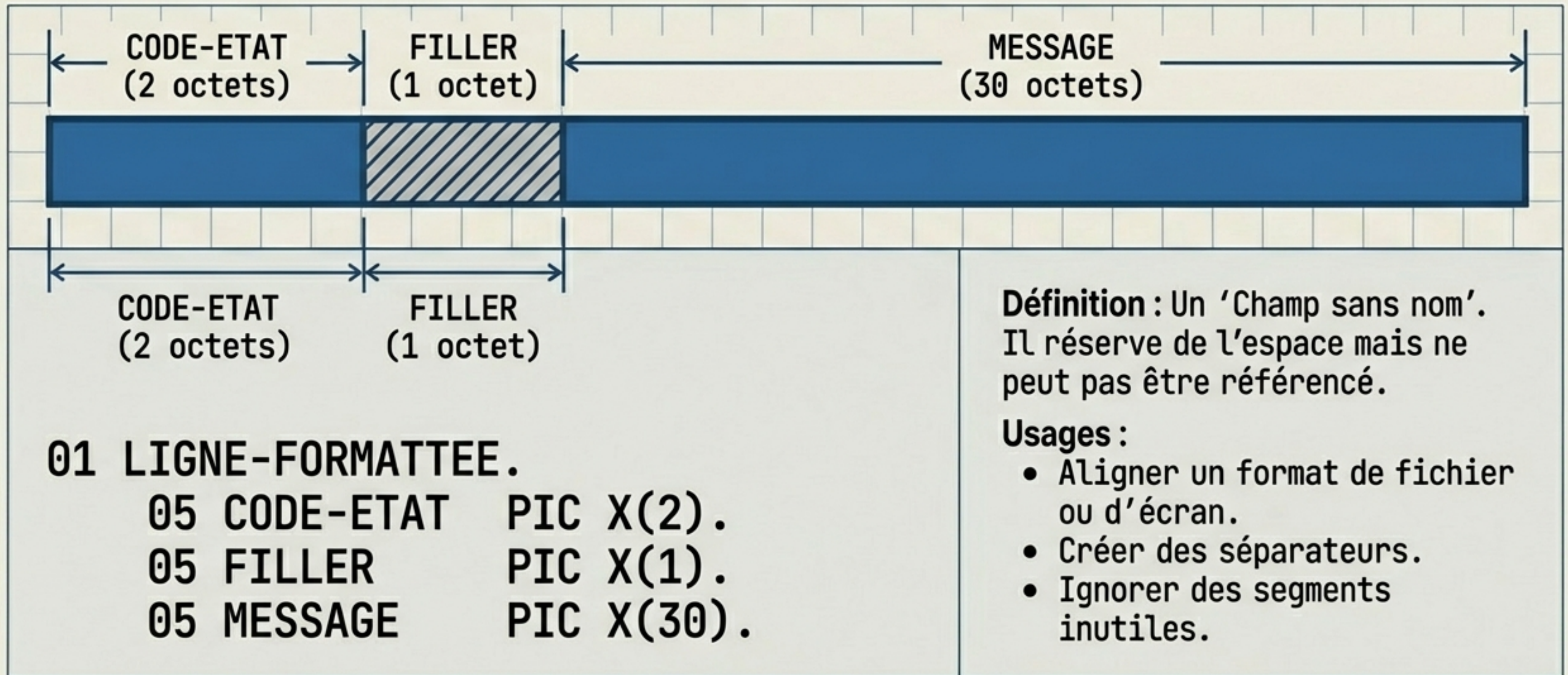
```
01 DOSSIER
  05 CODE
  05 LIBELLE
```



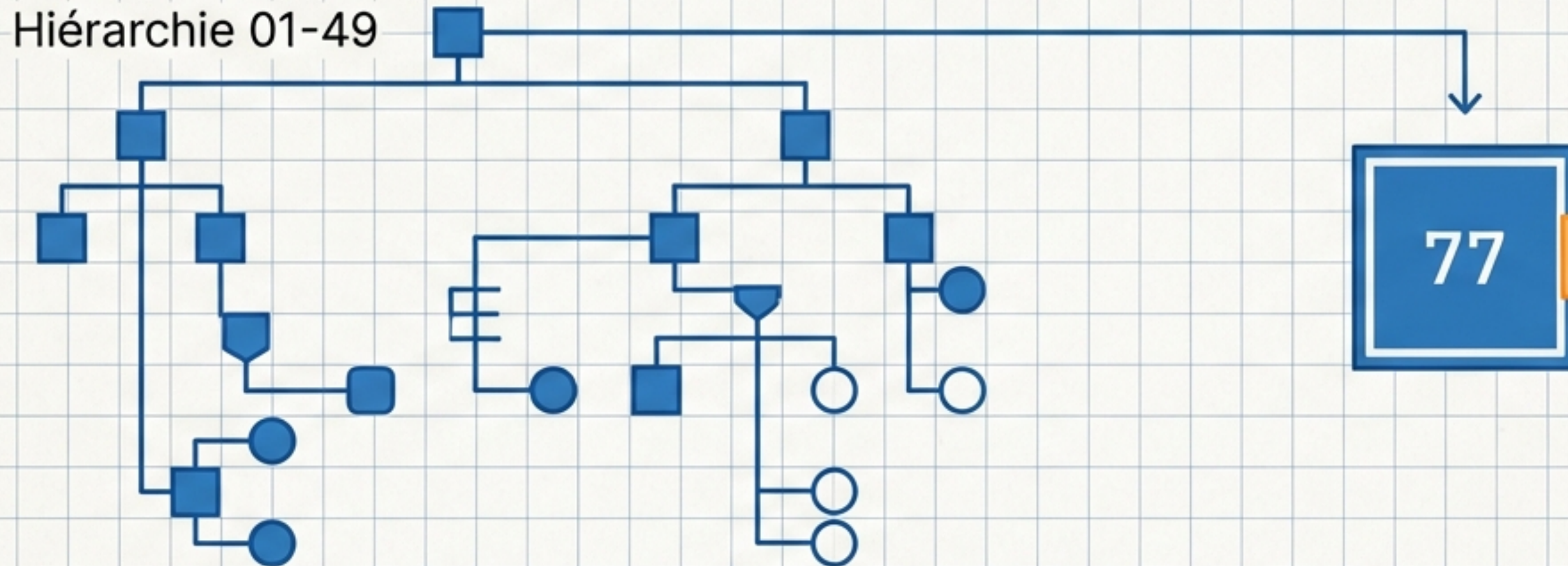
Sauter des numéros est permis (pas de niveau 10 ici). Les numéros indiquent la profondeur relative.

Bonne Pratique : Utilisez les intervalles standards pour la lisibilité (01 -> 05 -> 10).

Le Mot-clé FILLER



Niveau 77 : L'Indépendant



Concept : Une variable qui existe "seule", strictement en dehors de toute hiérarchie de Groupe 01.

77	TAUX-TVA	PIC 9V99	VALUE 0.20.
77	NB-ESSAIS	PIC 99	VALUE 0.

Contexte : Fréquent dans le code legacy. L'usage moderne tend à regrouper les variables sous des niveaux 01, mais le 77 reste valide pour des compteurs ou constantes isolés.

Niveau 88 : La Sémantique

Avant

Code Cryptique

IF CODE-STATUT = '0'



Après

Langage Naturel

IF STATUT-OK



Code :

```
01 CODE-STATUT      PIC X.  
   88 STATUT-OK      VALUE '0'.  
   88 STATUT-ERREUR  VALUE 'E'.
```

! Règle Cruciale :

Le niveau 88 n'est pas une variable, c'est une étiquette attachée à une variable parente. Elle agit comme un interrupteur booléen.

Niveau 88 : Listes et Plages de Valeurs

Example 1 : Listes (Grouping Values)

```
88 PAIEMENT-VALIDE VALUE 'C' 'V' 'T'.
```

Une condition peut couvrir plusieurs valeurs.

Usage :

```
IF PAIEMENT-VALIDE (Vrai si C, V, ou T).
```

Example 2 : Plages (THRU)

```
01 NOTE-SUR-100 PIC 999.  
88 NOTE-EXCELLENTE VALUE 085 THRU 100.  
88 NOTE-FAIBLE VALUE 000 THRU 059.
```

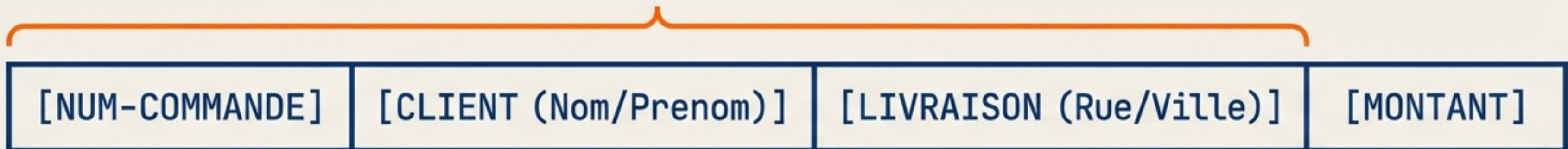
Usage :

```
IF NOTE-EXCELLENTE DISPLAY 'Bravo'.
```

Impact : Réduction drastique de la logique complexe (IF/OR) dans la Procedure Division.

Niveau 66 : La Vue Alternative (RENAMES)

66 ENTETE-COMMANDE



[NUM-COMMANDE]	[CLIENT (Nom/Prenom)]	[LIVRAISON (Rue/Ville)]	[MONTANT]
----------------	-----------------------	-------------------------	-----------

```
66 ENTETE-COMMANDE RENAMES NUM-COMMANDE THRU LIVRAISON.
```

Concept : Nommer une tranche contiguë de données qui traverse les frontières des groupes.

Règles Strictes :

- Doit être contigu (mémoire physique adjacente).
- Pas de clause PIC (hérite de la structure).
- Ne peut pas renommer les niveaux 01, 77 ou 88.

Synthèse : Le Système Complet

```
01 FICHE-PAIEMENT. ←  
    05 NUM-DOSSIER PIC 9(8).  
    05 IDENTITE.  
        10 NOM PIC X(20).  
    05 CODE-ETAT PIC X.  
        88 ETAT-ACCEPTE VALUE 'A'.  
    05 MONTANT-DU PIC 9(6)V99.  
66 ENTETE-PAIEMENT RENAMES NUM-DOSSIER THRU IDENTITE.  
77 NB-TRAITEMENTS PIC 99 VALUE 0. →
```

**Structure &
Hiérarchie**

**Logique
Sémantique**

Vue Alternative

**Compteur
Indépendant**

Procedure :

```
IF ETAT-ACCEPTE  
    DISPLAY ENTETE-PAIEMENT
```

Les Pièges à Éviter



Niveau ≠ Taille

Le numéro indique la position, pas la taille. La taille est déterminée exclusivement par PIC.



L'Illusion du Groupe

Un Groupe (05) ne stocke pas de données supplémentaires. Sa longueur est simplement la somme de ses enfants.



Dépendance du 88

Un 88 ne peut pas être "défini" directement. Vous devez faire un MOVE vers la variable parente pour changer l'état du 88.



Complexité du 66

N'utilisez pas 66 comme un simple alias. Utilisez-le uniquement pour regrouper une plage contiguë de champs.

Méthode de Progression

Les Bases

Commencez par un simple
01 et deux **05**.

Sémantique

Ajoutez un champ code et
définissez des conditions **88**.
Écrivez des tests **IF** lisibles.

Analyse

Lisez du code existant :
identifiez les 'Racines' (**01**)
et les 'États' (**88**).

1

2

3

4

5

Structure

Créez un Groupe (**05**) contenant
des éléments (**10**).
Observez comment **DISPLAY**
concatène le tout.

Expérimentation

Essayez un niveau **66**
Essayez un niveau **66** sur un
segment contigu pour comprendre
le concept de '**tranche**'.

Structurez pour Mieux Régner



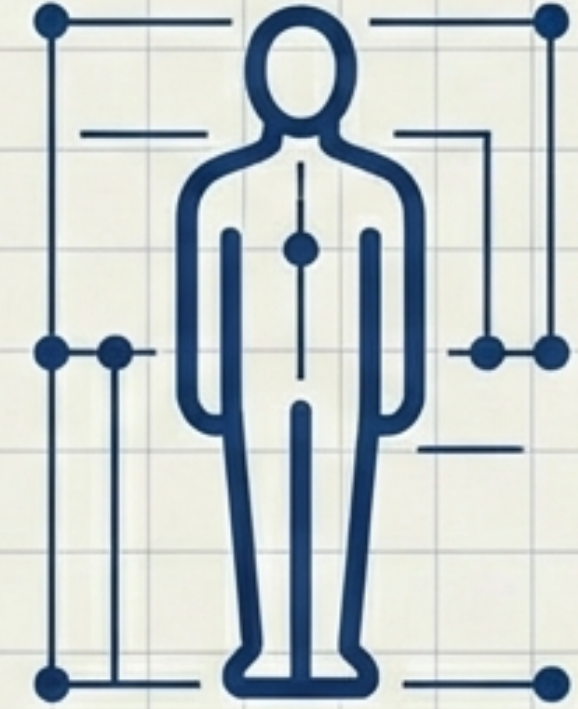
01-49
(Structure)



88
(Sens)



66
(Vue)



77
(Solo)

**Les niveaux COBOL ne sont pas de simples contraintes syntaxiques ;
ce sont un puissant système de modélisation. Les maîtriser vous
permet de contrôler les données avec précision et clarté.**