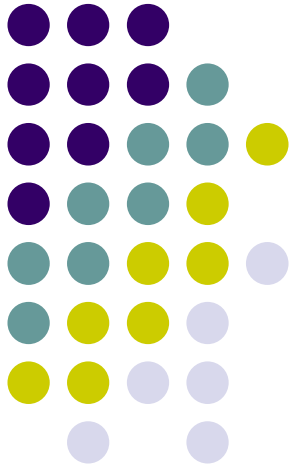
Programming Whitout Code : An Approach & Environment To Conditions-on-Data Programming

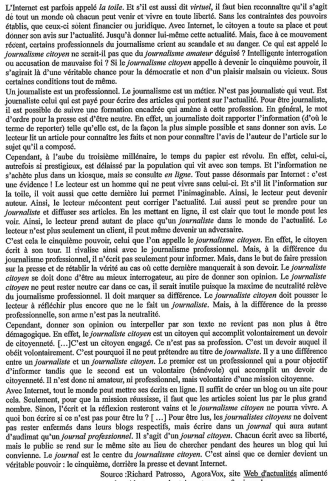
Ph. Larvet

Former Research Engineer,
Alcatel-Lucent Bell Labs

June 2017



I Have A Dream



L'Internet est parfois appelé *la toile*. Et s'il est aussi dit *virtuel*, il faut bien reconnaître qu'il s'agit de tout un monde où chacun peut venir et vivre en toute liberté. Sans les contraintes des pouvoirs établis, que ceux-ci soient financier ou juridique. Avec Internet, le citoyen a toute sa place et peut donner son avis sur l'actualité. Jusqu'à donner lui-même cette actualité. Mais, face à ce mouvement récent, certains professionnels du journalisme crient au scandale et au danger. Ce qui est appelé le *journalisme citoyen* ne serait-il pas que du *journalisme amateur* déguisé ? Intelligente interrogation ou accusation de mauvaise foi ? Si le *journalisme citoyen* appelle à devenir le cinquième pouvoir, il s'agirait là d'une véritable chance pour la démocratie et non d'un plaisir malsain ou vicieux. Sous certaines conditions tout de même.

Un journaliste est un professionnel. Le journalisme est un métier. N'est pas journaliste qui veut. Est journaliste celui qui est payé pour écrire des articles qui portent sur l'actualité. Pour être journaliste, il est possible de suivre une formation encadrée qui amène à cette profession. En général, le mot d'ordre pour la presse est d'être neutre. En effet, un journaliste doit rapporter l'information (d'où le terme de reporter) telle qu'elle est, de la façon la plus simple possible et sans donner son avis. Le lecteur lit un article pour connaître les faits et non pour connaître l'avis de l'auteur de l'article sur le sujet qu'il a composé.

Cependant, à l'aube du troisième millénaire, le temps du papier est révolu. En effet, celui-ci, autrefois si prestigieux, est délaissé par la population qui vit avec son temps. Et l'information ne s'achète plus dans un kiosque, mais se consulte *en ligne*. Tout passe désormais par Internet : c'est une évidence ! Le lecteur est un homme qui ne peut vivre sans celui-ci. Et s'il lit l'information sur la toile, il voit aussi que cette dernière lui permet l'inimaginable. Ainsi, le lecteur peut devenir auteur. Ainsi, le lecteur mécontent peut corriger l'actualité. Lui aussi peut se prendre pour un *journaliste* et diffuser ses articles. En les mettant en ligne, il est clair que tout le monde peut les voir. Ainsi, le lecteur prend autant de place qu'un *journaliste* dans le monde de l'actualité. Le lecteur n'est plus seulement un client, il peut même devenir un adversaire.

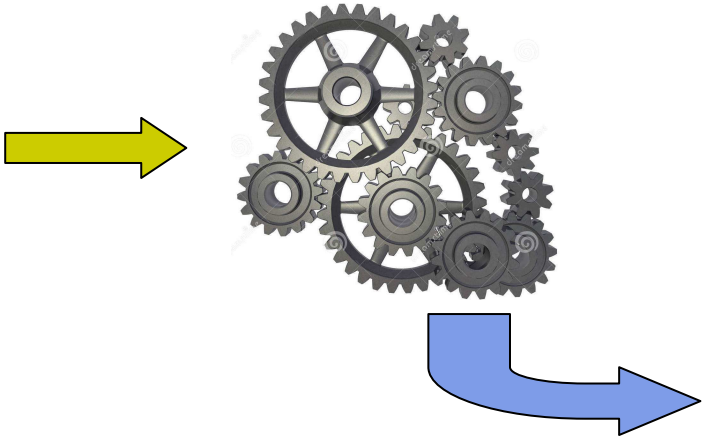
C'est cela le cinquième pouvoir, celui que l'on appelle le *journalisme citoyen*. En effet, le citoyen écrit à son tour. Il rivalise ainsi avec le journalisme professionnel. Mais, à la différence du journalisme professionnel, il n'écrit pas seulement pour informer. Mais, dans le but de faire pression sur la presse et de rétablir la vérité au cas où cette dernière manquerait à son devoir. Le *journaliste citoyen* se doit donc d'être au mieux interrogateur, au pire de donner son opinion. Le *journaliste citoyen* ne peut rester neutre car dans ce cas, il serait inutile puisque la maxime de neutralité relève du journalisme professionnel. Il doit marquer sa différence. Le *journaliste citoyen* doit pousser le lecteur à réfléchir plus encore que ne le fait un *journaliste*. Mais, à la différence de la presse professionnelle, son arme n'est pas la neutralité.

Cependant, donner son opinion ou interpeller par son texte ne revient pas non plus à être démagogique. En effet, le *journaliste citoyen* est un citoyen qui accomplit volontairement un devoir de citoyenneté. […]C'est un citoyen engagé. Ce n'est pas sa profession. C'est un devoir auquel il obéit volontairement. C'est pourquoi il ne peut prétendre au titre de *journaliste*. Il y a une différence entre un *journaliste* et un *journaliste citoyen*. Le premier est un professionnel qui a pour objectif d'informer tandis que le second est un volontaire (bénévole) qui accomplit un devoir de citoyenneté. Il n'est donc ni amateur, ni professionnel, mais volontaire d'une mission citoyenne.

Avec Internet, tout le monde peut mettre ses écrits en ligne. Il suffit de créer un blog ou un site pour cela. Seulement, pour que la mission réussisse, il faut que les articles soient lus par le plus grand nombre. Sinon, l'écrit et la réflexion resteront vains et le *journalisme citoyen* ne pourra vivre. A quoi bon écrire si ce n'est pas pour être lu ? […] Pour être lus, les *journalistes citoyens* ne doivent pas rester enfermés dans leurs blogs respectifs, mais écrire dans un *journal* qui aura autant d'audimat qu'un *journal professionnel*. Il s'agit d'un *journal citoyen*. Chacun écrit avec sa liberté, mais le public se rend sur le même site au lieu de chercher pendant des heures un blog qui lui convienne. Le *journal* est le centre du *journalisme citoyen*. C'est ainsi que ce dernier devient un véritable pouvoir : le cinquième, derrière la presse et devant Internet.

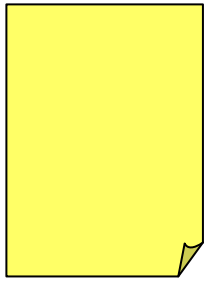
Source :Richard Patrosso, AgoraVox, site Web d'actualités alimenté par des rédacteurs volontaires et non professionnels.

Specifications,
expressed in Natural Language

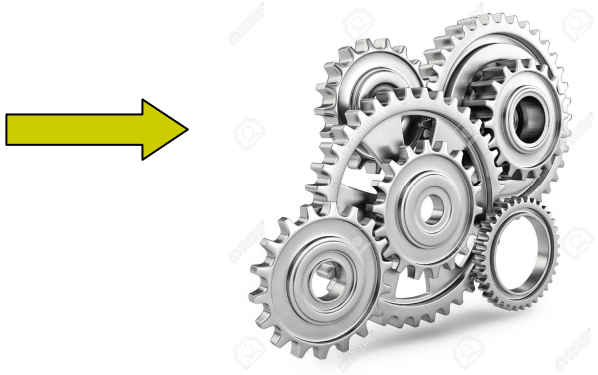


Generated code

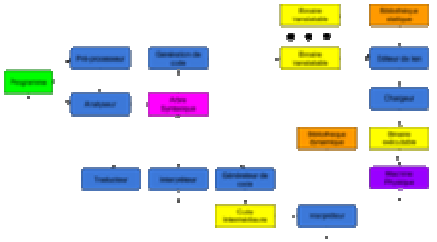
The Present Approach



Spec of a complex system
to be validated

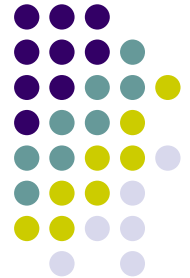


Mechanism for validation
or simulation

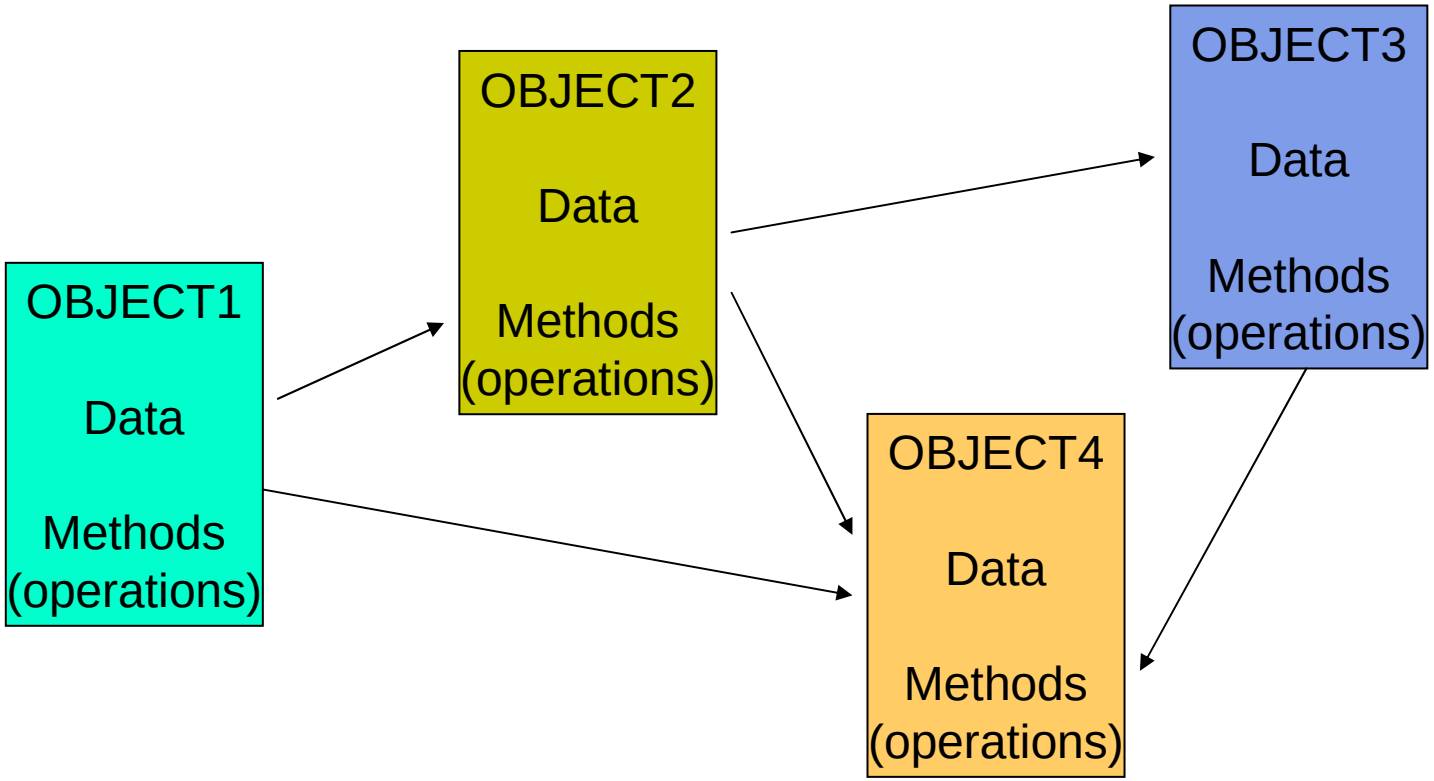


= A Program!

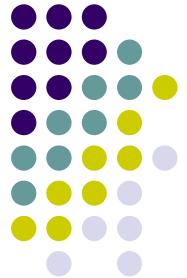
What Is A Program ?



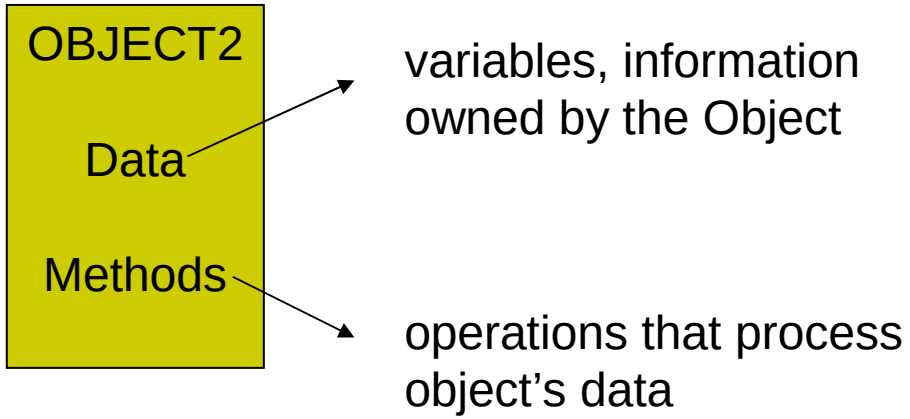
- Within the context of OO programming, a Program is a set of interacting Objects



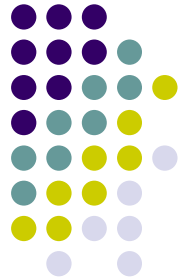
What is an Object ?



- An Object is a code structure that encapsulates **data** and **methods** that work on them

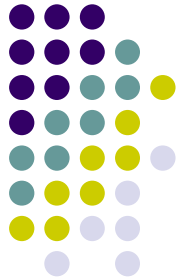


Methods Contain Statements



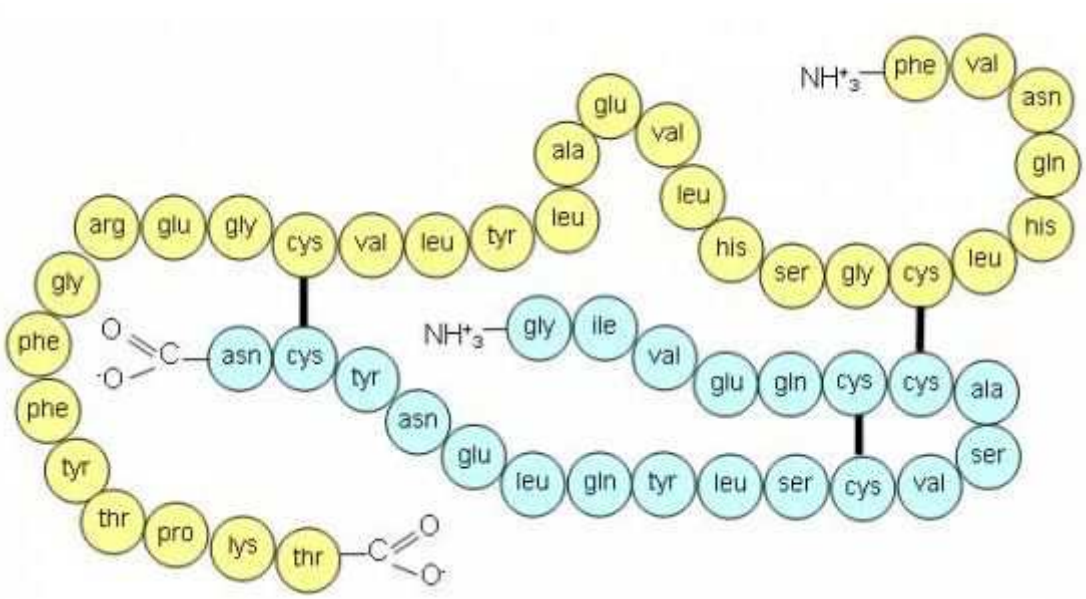
- Statements (instructions) can be divided in four main groups:
  - **A**ccessors = to get data from other objects
  - **C**alculations = to compute new values of variables & data
  - **T**ests = to organize the internal logic of the method (if… then… else; do… loop; while…; repeat... until; etc.)
  - **G**enerators of events = to call methods on other objects

The Bricks Of A Method



- => A, T, C, G are fundamental bricks to build a method
- Do these letters remind you something?
- (Let’s open a small parenthesis)

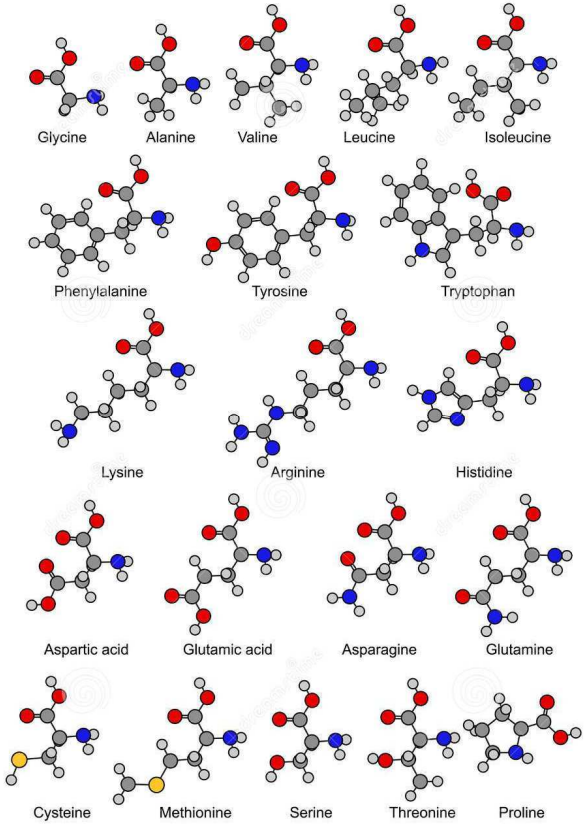
A Protein Is Made Of Bricks

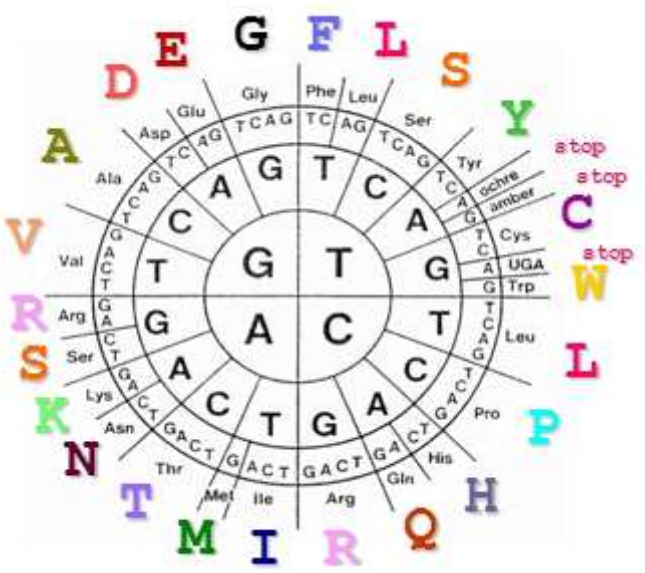


Sample : amino-acids sequence in human insulin

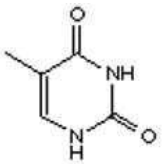
Final Bricks Are A,T,C,G

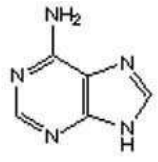
- Each amino-acid (Gly, Val, Ala, Glu, Pro, etc.) is a set of A,C,T,G

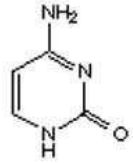


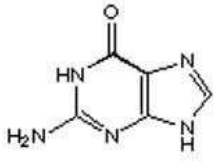


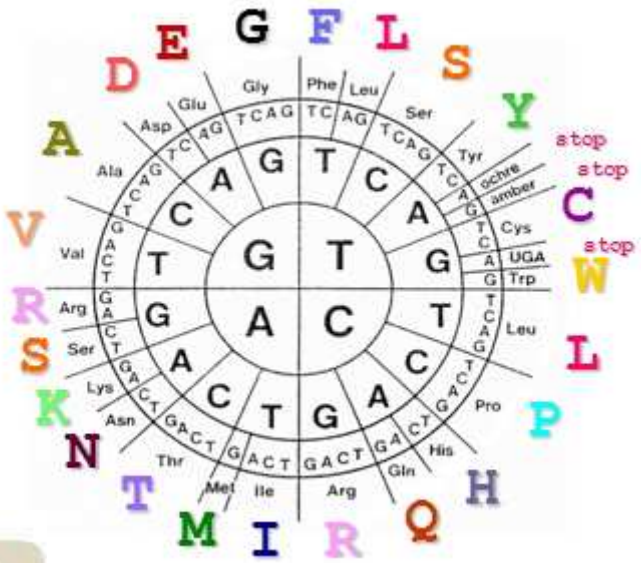
A,T,C,G : The Bricks of Life

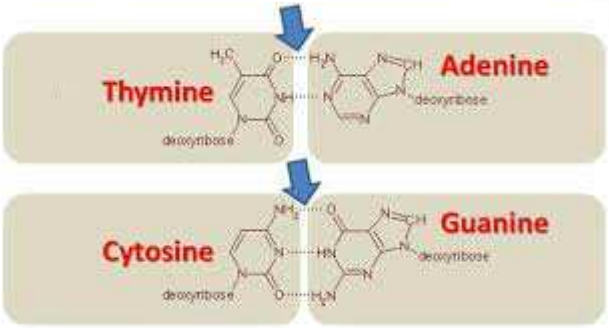


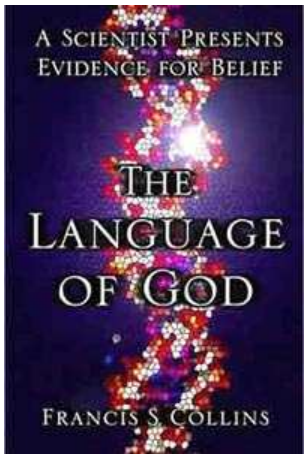


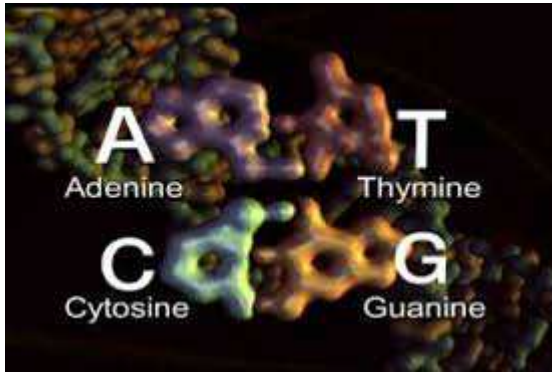




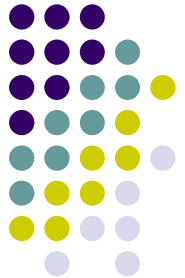








A,C,T,G To Build Programs

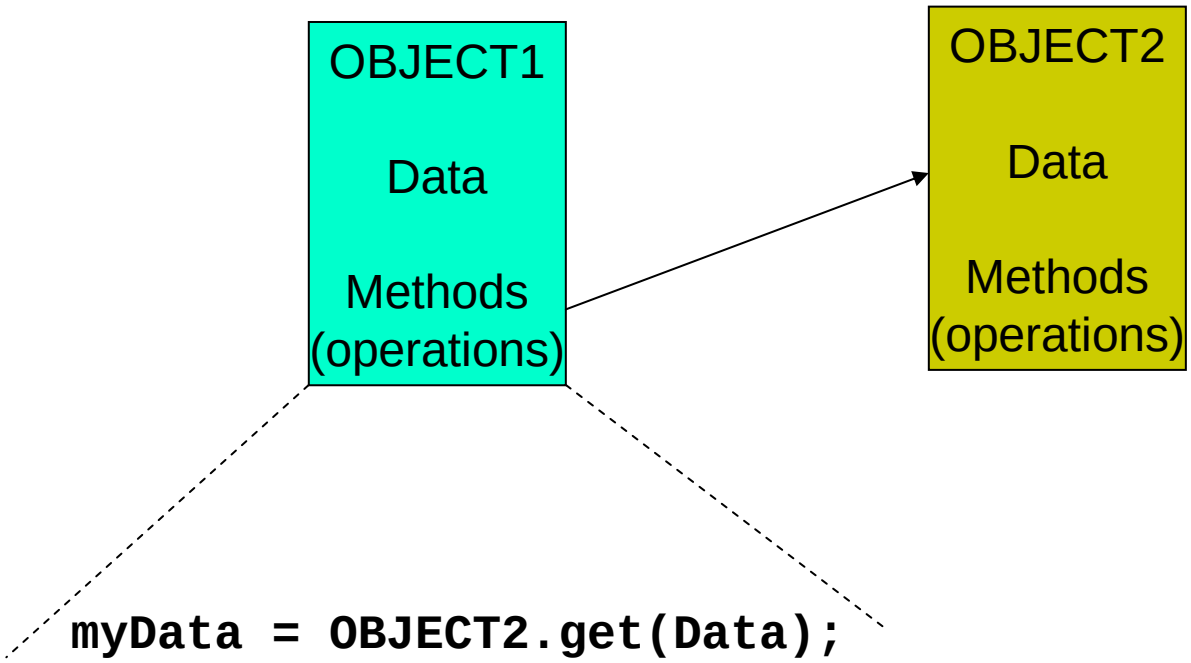


- As a Protein is a Program, a Program can be built as a set of A,C,T,G

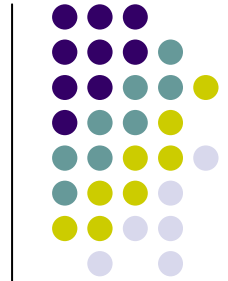
  (Let’s close now the small parenthesis)

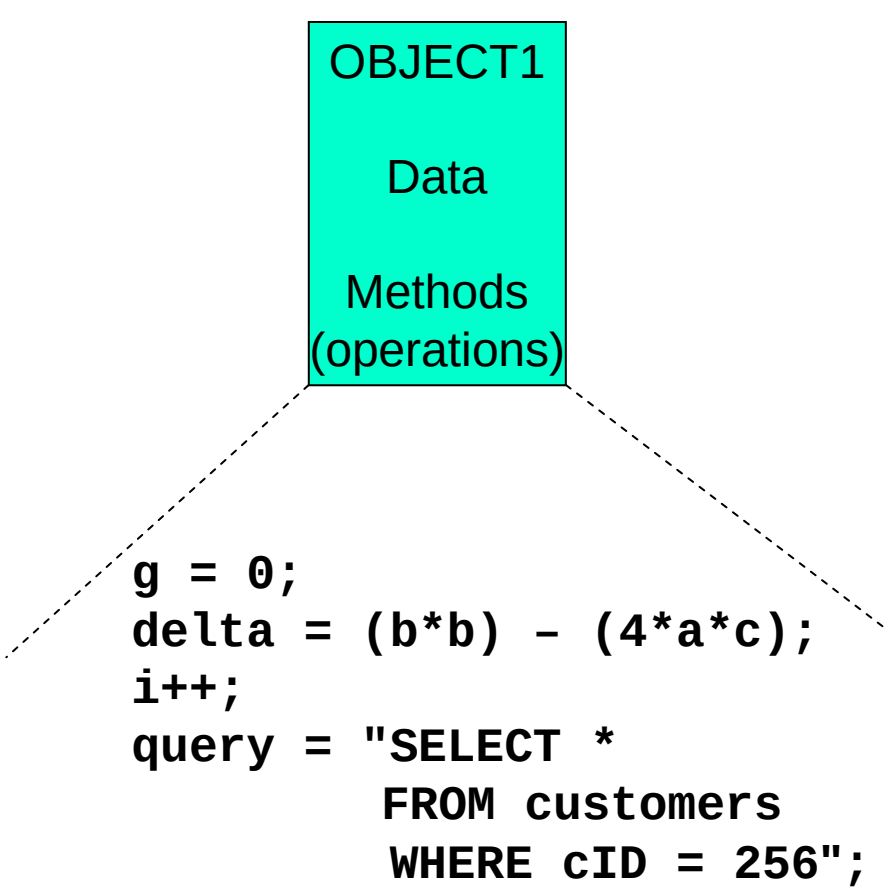
- From a detailed point of view, A,C,T,G can be seen as equalities (equations) or « conditions on data » (COD)

An Accessor Is An Equality

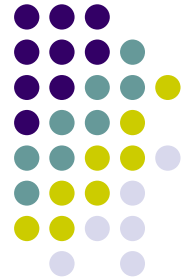


A Calculation Is An Equality

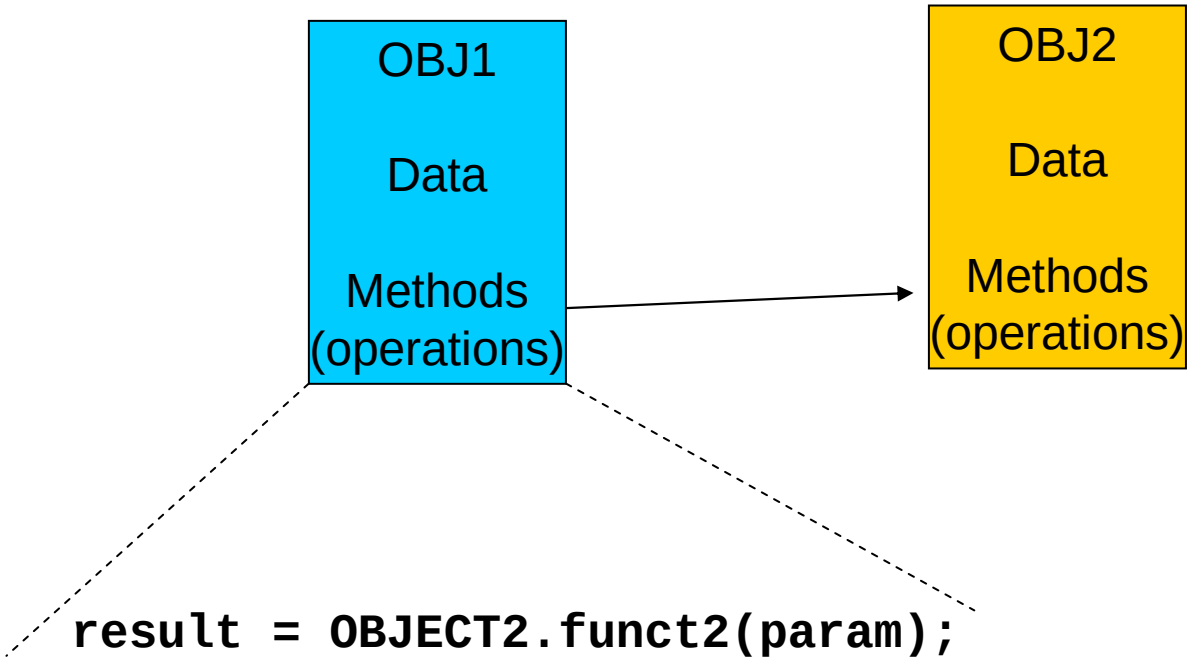




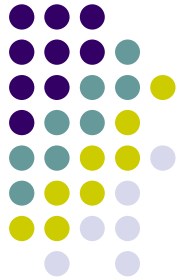
A Generator Is An Equality

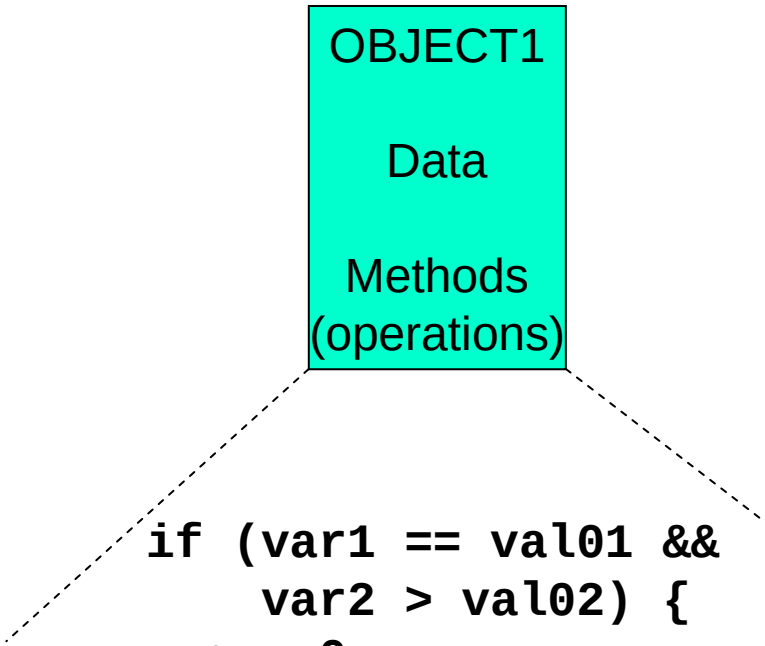


- An event Generator inside a method of OBJ1 is a call of a method on OBJ2



A Test Is A Condition On Data



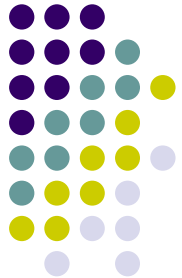


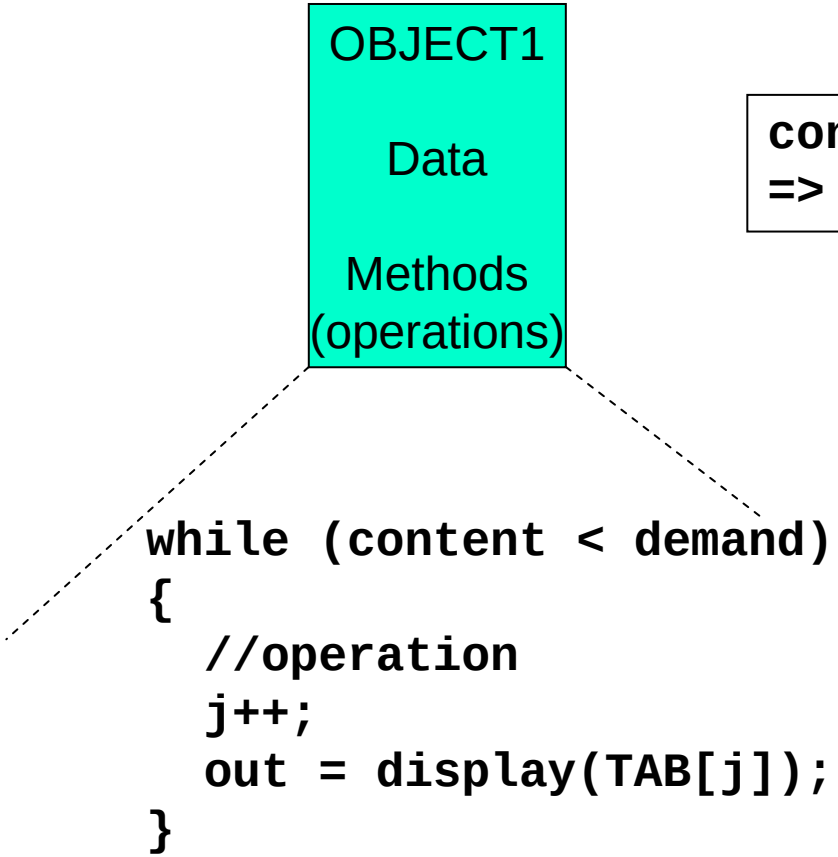

```
if (var1 == val01 &&
    var2 > val02) {
  g = 0;
  i++;
}
```

can be expressed as :

```
var1 = val01 AND var2 > val02
=> g = 0 AND i++
```

A Test Is A Condition On Data (2)



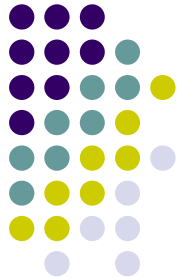


can be expressed as :

```
content < demand
=> j++ AND out = display(TAB[j])
```

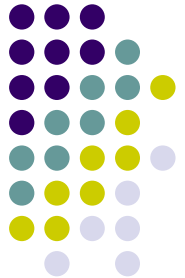
and we need a mechanism to express or implement the boucle

First Summary



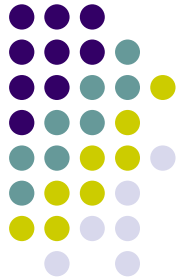
- We propose to simplify the writing of programs by expressing only
  - equalities
  - conditions on data (COD)
- So, we don’t write any more complex algorithms in terms of « code », but only CODs
- And we need a mechanism to implement the dynamics => HOW TO DO THIS ?

A COD Is A Rule



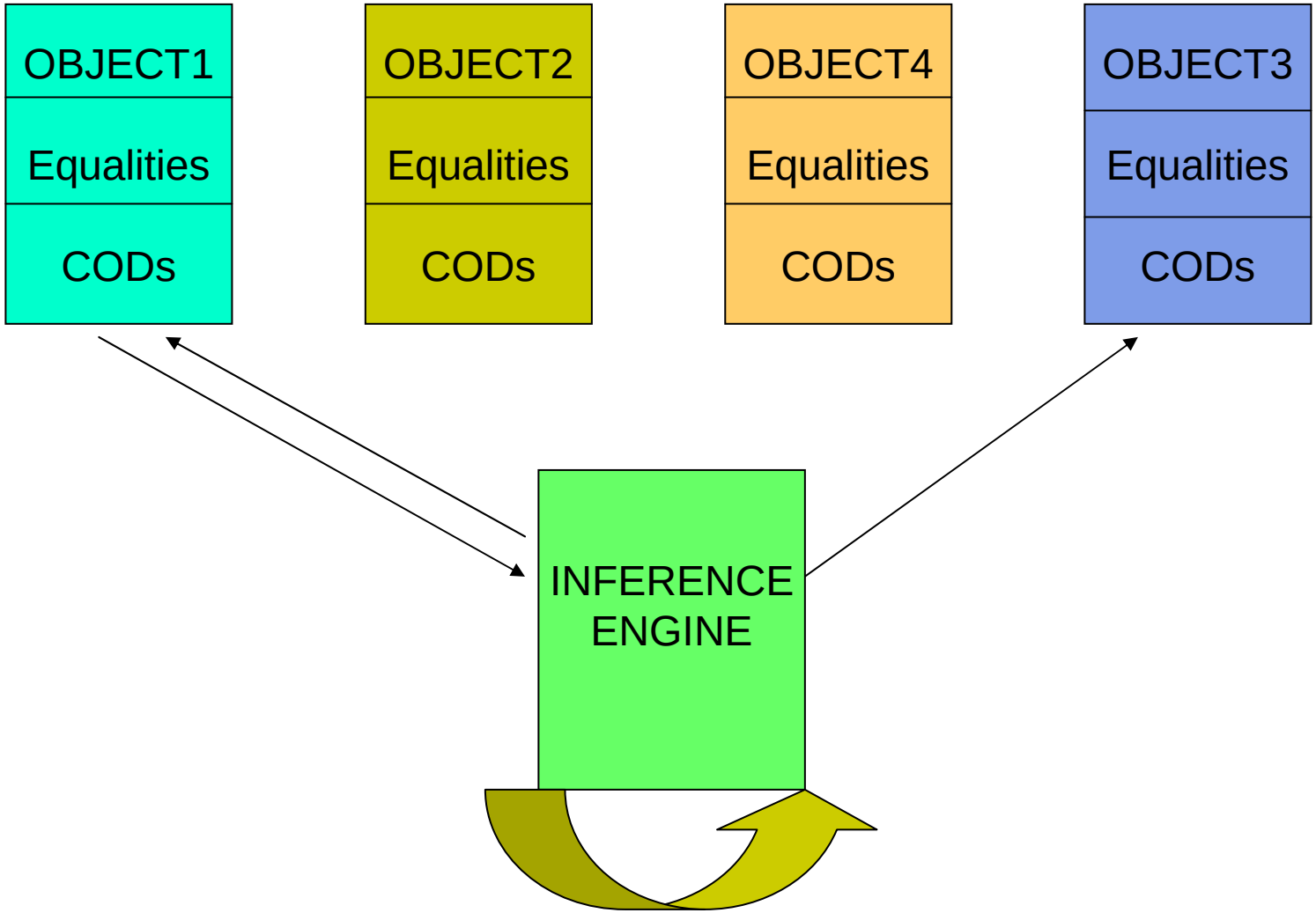
- A condition on data, expressed as
  - **`var1 = val01 => g = 0 AND i++`**
- can be seen as a « production rule » in a rule-based system (expert system) :
  - PREMISE : var1 = val01
  - CONCLUSION : g = 0 AND i = i +1

CODs Are Distributed Rules

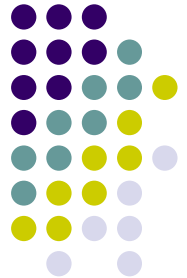


- Main differences between a rule-based system and the COD approach :
  - in our COD approach, CODs express conditions on equalities (i.e. equations) instead of predicates or « logical production rules »
  - rules (i.e. CODs) are not stored in a unique **rule database** but are ***distributed*** into all objects because a COD belongs to a given object
  - idem, facts (i.e. Data) are not stored in a unique **fact database** but are ***distributed*** into objects, because each object stores its own data

The COD Environment



Inside The Inference Engine



```
'on balaie les "faits" de TabN
  'et on cherche dans les "règles" des objets si certaines prémisses peuvent décl. des concl.
  'on répète le traitement jusqu'à ce qu'aucune nouvelle concl ne soit décl.

  nouv_conclusion = False

    'on balaie les RLists des objets pour chercher une éventuelle règle à déclencher
    For i = 0 To nbObj
      'on balaie la RList de chaque objet
      For j = 0 To RList(i).ListCount - 1
        reg$ = RList(i).List(j)

        prem$ = gauche(reg$, "=>") 'extraction de la prémisse
        If InStr(prem$, " AND ") > 0 Then
          sprem1$ = gauche(prem$, " AND ")
          sprem2$ = droite(prem$, " AND ")
          If prem_verifiee(sprem1$) = True And prem_verifiee(sprem2$) = True Then
            decl_concl i, j
            nouv_conclusion = True
          End If

        Else
          If prem_verifiee(prem$) = True Then
            decl_concl i, j
            nouv_conclusion = True
          End If
        End If

      Next
    Next

  If nouv_conclusion = True Then moteur
```

Building An Example

Demo !

Object Model of the Example