



# **Analysis of the Designs of Coupling Measures : A Case Study**



**Cheikhi, Abran & Lopez**



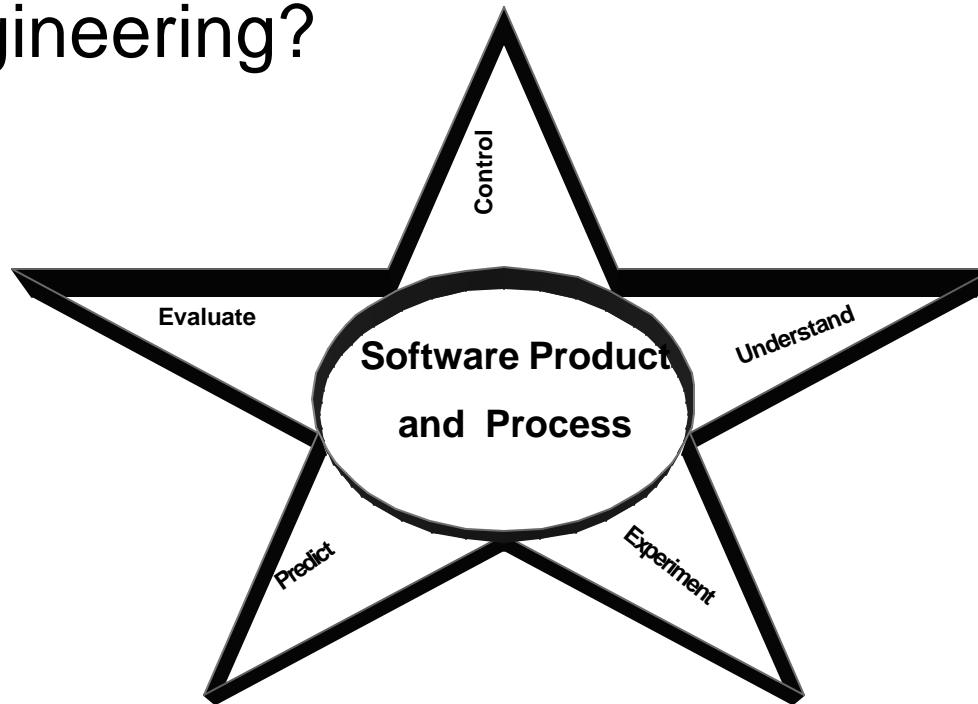
# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work



# *Motivations*

- Why the measure is important in Software Engineering?



*Pfleeger and Oman (1997)*



# *Motivations*

- In Software Engineering, various measures have been proposed
- To evaluate the quality of the design of the software such as coupling, many coupling measures and models have been proposed
- Diversity in measures → Variety of objectives and corresponding designs



## *Motivations*

- What has been missing is a proper and an exhaustive investigation of :
  - *The designs of the coupling measures*
  - *The models and the classifications of coupling*
  
- Need of a reference framework on coupling concept to consolidate the suggested coupling measures



# *Objectives*

- Investigate and clarify the design of the
  - Coupling Between Objects (CBO) measure proposed by *Chidamber and Kemerer*
  - based on *Abran and Jacquet's* model of the process for designing a measurement method
- Point out some weaknesses of the methods used up to now in the design of measures and to identify their deficiencies
- Propose an initial framework for coupling based on the existing ones



# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work



# *High level Model-Design Measure*

## **Measures problems**

- Different approaches to software measurement validation
- Jumping into the proposition of the measures without properly defining what they are trying to measure
- Ambiguity in the definition of the concept to be measured  
:Different applications of the measure → Different results →  
Different interpretations of the results
- .....





# *High level Model-Design Measure*

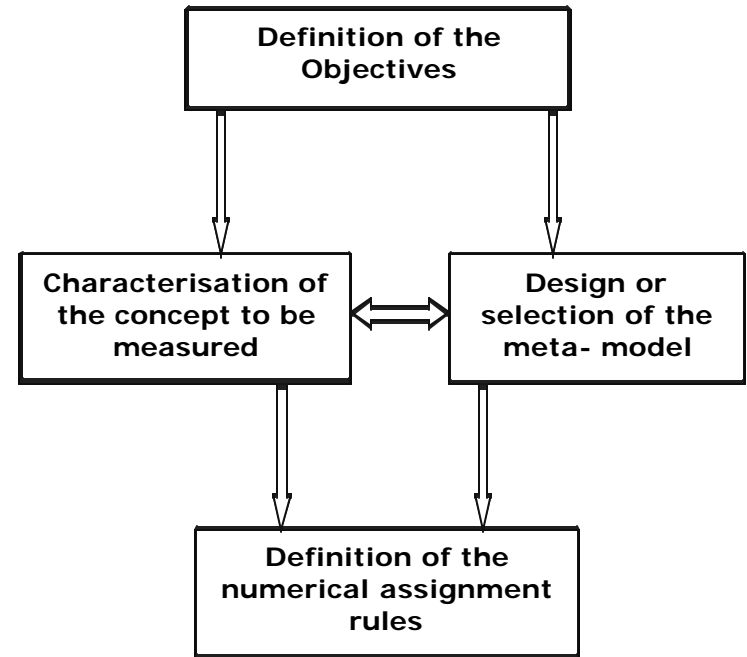
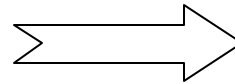
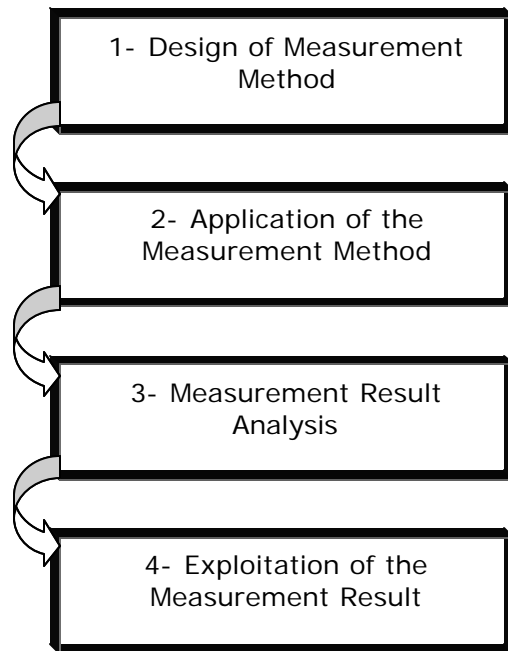
## **Abran and Jacquet's (1997) High level Measurement process Model allowed:**

- to properly state the objective of the measure before going any further
- to identify the characteristics of the measured attribute before defining numerical assignment rules
- to think about the intended use of the measure before jumping on its application on the software
- to avoid ambiguity in the use of the measure and in the interpretation of the results of its application
- to easily analyse and exploit the results in multiple models



# Measurement Process Model

A Broader Measurement Process Model in Software Engineering



Abran and Jacquet (1997)



# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work



# Coupling

- What is Coupling ?
- Various views, various definitions
  - *“The measure of the strength of association established by a connection from one module to another”* Stevens et al. (1974)
  - *“The manner and degree of interdependence between software modules”* IEEE 610.12 (1990)
  - *“The degree of interdependence among the components of a software system”* Briand et al. (1997)



# Coupling Models

- Different types of coupling, different models
  - *Model of coupling based on six characteristics of coupling* - IEEE 610.12 (1990)
  - *Classification of coupling based on three categories and their related sub types* - Eder et al. (1994)
  - *Set of descriptive criteria for coupling but no General framework for coupling* - Briand et al. (1999)



# Coupling Measures

| MEASURES                          | DEFINITIONS   |
|-----------------------------------|---|
| Chidamber and Kemerer (1991,1994) |   |
| CBO                               | (Coupling Between Objects) :CBO for a class is a count of the number of other classes to which it is coupled.   |
| CBO'                              | The same definition, except that CBO for a class is a count of the number of non-inheritance related couples with other classes.  |
| RFC                               | (Response Set for Class): $RFC =  RS $ where RS is the response set for the class. $RS = \{Mi\} \cup \text{all } \{Ri\}$ , where $Mi$ = all methods in the class and $Ri$ = set of methods called by $Mi$ .   |
| Li and Henry (1993)               |   |
| MPC                               | (Message Passing Coupling): the number of send statements defined in a class.   |
| DAC                               | (Data Abstract Coupling): number of ADT's defined in a class. A variable declared within a class X may have a type of ADT which is another class definition. The numbers of attributes in a class that have type another class.   |
| Briand et al. (1997)              |   |
| IFCAIC/FCAEC                      | <p>These coupling measures are counts of interactions between classes. The measures distinguish the relationship between classes (friendship, inheritance, none), different types of interactions, and the locus of impact of the interaction.</p> <p>The acronyms for the measures indicates what interactions are counted:</p> <ul style="list-style-type: none"> <li>The first two letters indicate the relationship (A: coupling to ancestor classes, D: Descendents, F: Friend classes, IF: Inverse Friends (classes that declare a given class c as their friend), O: Others, i.e., none of the other relationships).</li> <li>The next two letters indicate the type of interaction:                             <ul style="list-style-type: none"> <li>CA: There is a Class-Attribute interaction between classes c and d, if c has an attribute of type d.</li> <li>CM: There is a Class-Method interaction between classes c and d, if class c has a method with a parameter of type class d.</li> <li>MM: There is a Method-Method interaction between classes c and d, if c invokes a method of d, or if a method of class d is passed as parameter (function pointer) to a method of class c.</li> </ul> </li> <li>The last two letters indicate the locus of impact:                             <ul style="list-style-type: none"> <li>IC: Import coupling, the measure counts for a class c all interactions where c is using another class.</li> <li>EC: Export coupling: count interactions where class d is the used class.</li> </ul> </li> </ul> |
| ACAIC/ DCAEC                      |   |
| OCAIC/OCAEC                       |   |
| IFCMIC/FCMEC                      |   |
| ACMIC/DCMEC                       |   |
| OCMIC/OCMEC                       |   |
| OMMIC/OMMEC                       |   |
| IFMMIC/FMMEC                      |   |
| AMMIC/DMMEC                       |   |



# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work



## *Case Study*

- Analyse the design of the Coupling Between Objects measure (CBO) proposed by *Chidamber and Kemerer*
- Use *Abran and Jacquet's* process for designing a measurement method
- Verify whether or not the CBO measure clearly refers to the design concepts.





# Case Study

- Chidamber and Kemerer based the definition of the Coupling Between Objects Measure on:

*“an object is coupled to another object if one of them acts on the other, i.e., methods of one use methods or instance variables of another. Since objects of the same class have the same properties, two classes are coupled when methods declared in one class use methods or instance variables defined by the other class”*

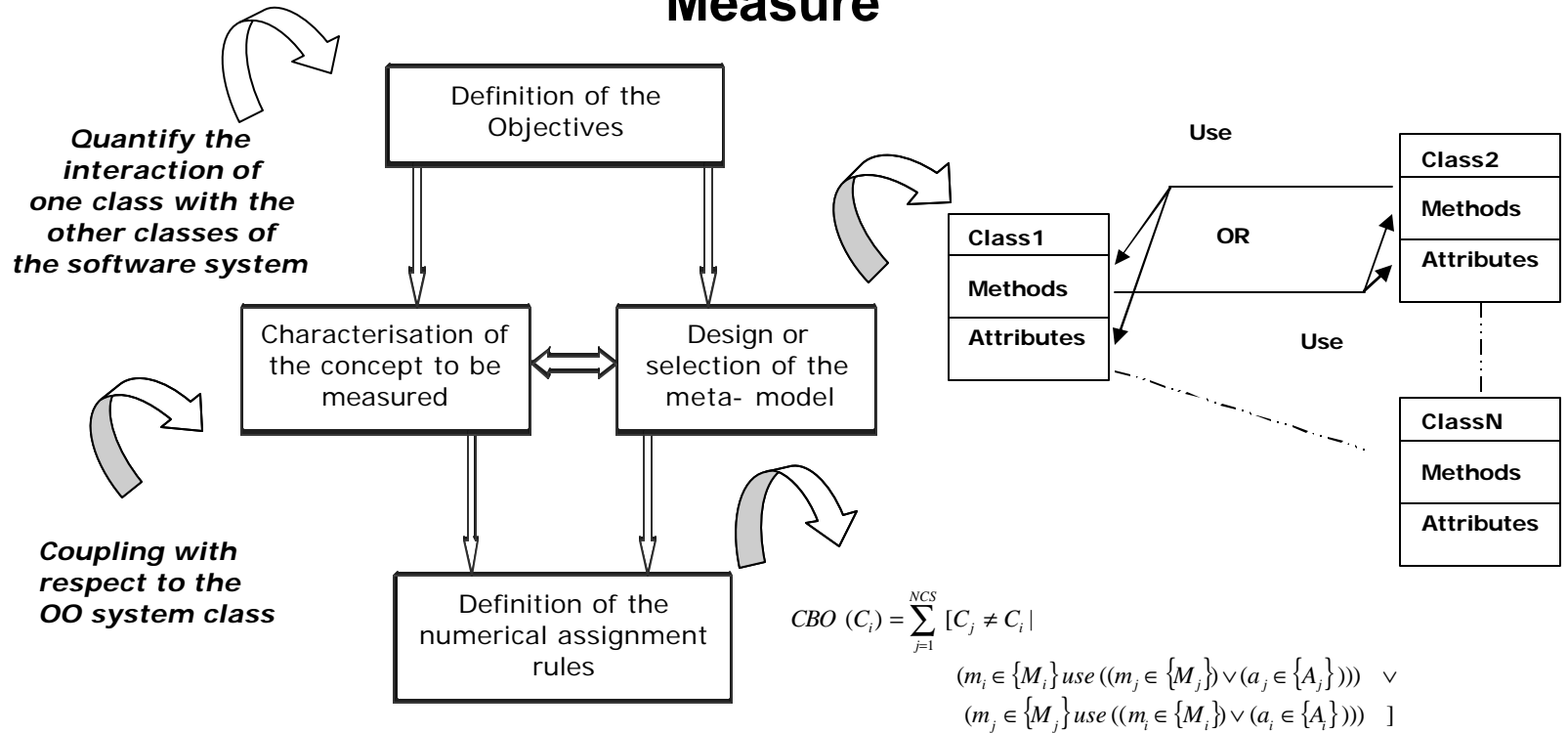
- Definition of CBO:

*“CBO for a class is a count of the number of other classes to which it is coupled”*  
Chidamber and Kemerer (1994)



# Case Study

## Detailed Model of the design of measurement method for CBO Measure





# Case Study- Summary

- Chidamber and Kemerer CBO measure
  - The coupling is defined as a relation
  - The purpose is to count the number of classes to which the measured class is coupled
  - The unit of measurement is the class
  
- But no direct information on
  - The attribute to be measured
  - The meta-model
  - The numerical assignment rules



# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work

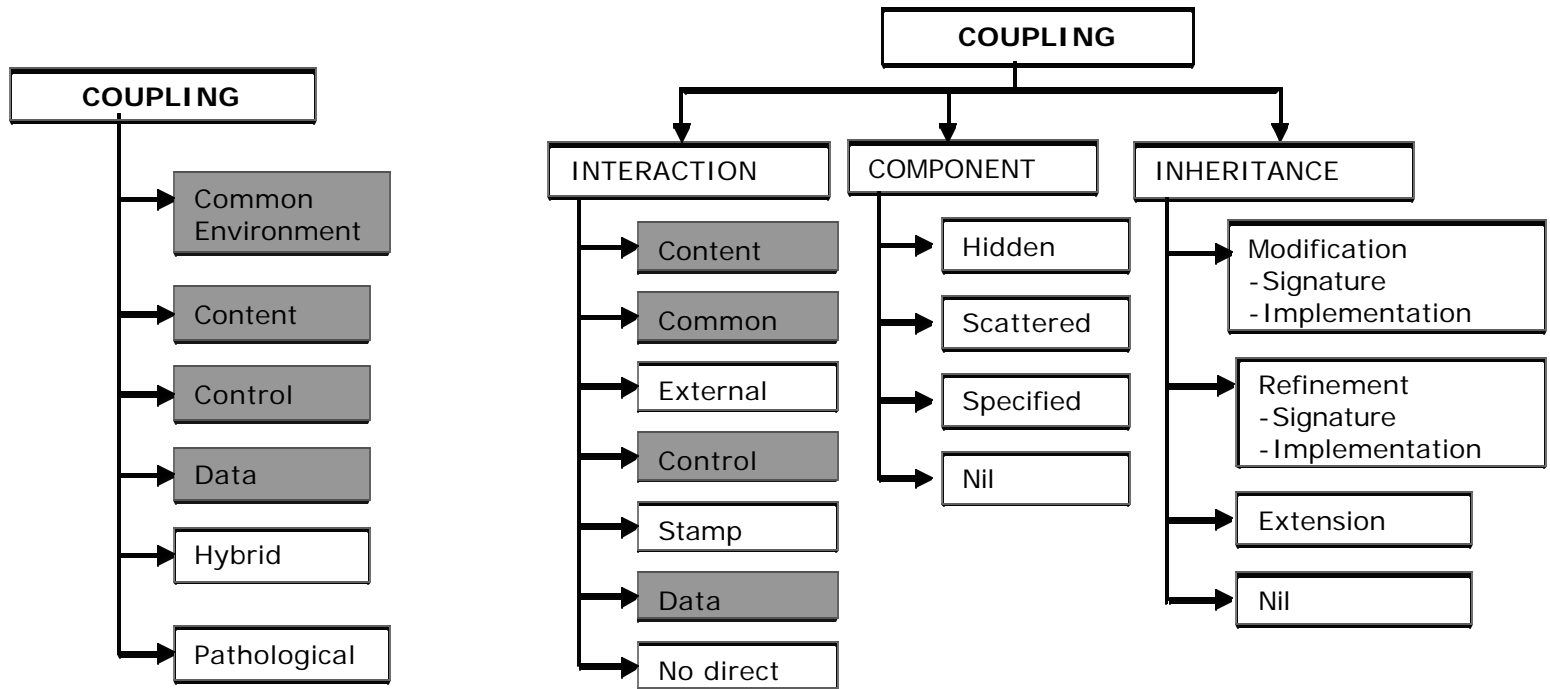


# *High level Model for coupling*

- In SE, various coupling views, various types of coupling, various models ( IEEE 610.12 -1990, Eder et al. -1994,.....)
- None of which has been used as reference by software engineers
- Need of a reference framework



# High level Model for coupling



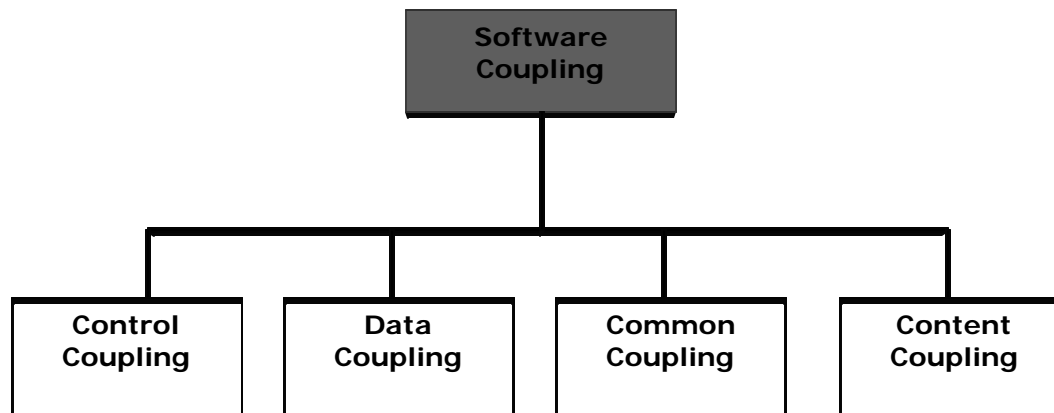
IEEE 610.12 (1990)

Eder et al. (1994)



# *High level Model for coupling*

## **Proposed Coupling model**





# *High level Model for coupling-Summary*

**To establish a successful High level Model of Coupling will require to:**

- Agree on the definition of coupling
- Conduct an in-depth study of all the coupling models suggested by the researchers in Software Engineering
  
- Establish the High level model : General framework of coupling
- Gain a consensus among the researchers of this framework
  
- Study the designs of the existing coupling measurements and integrate them into its corresponding category in the model
  
- Use the framework as a basis for:
  - the definition of new measures by researchers
  - the choice of them by users.





# *Agenda*

- Motivations and Objectives
- High level Model- Design of Measure
- Coupling
- Case Study: Coupling Between Objects (CBO) measure
- High level Model for coupling
- Conclusions and Future Work



# Summary & Conclusion

- **Chidamber and Kemerer's Coupling Between Objects (CBO) Measure**
  - Quantify the interaction of one class with the other classes of the software system
  - Does not cover directly the design of the measurement method (meta model, numerical rules)
  
- **Abran and Jacquet's process Model**
  - Avoid ambiguity in the definition of the new measures
  - Clarify and point out weaknesses in the design of the measures
  - Identify the deficiencies in the existing ones
  - Propose improvements



# *Summary & Conclusion*

- **State of the art of coupling concept**
  - Rich body of definitions, models or classifications and measures
  - But no consensus
  
- **A Standardized framework for coupling concept would provide researchers with:**
  - Clear definitions
  - Understandable practices
  - Adequate and unambiguous use of the rules at the time of the design of new coupling measurements



## *Conclusions Cont.*

### ○ **Proposed Coupling Model**

- Interesting initial work
- But incomplete

### ○ **Future Work**

- 1<sup>st</sup> step: Basic study of the existing coupling models
- 2<sup>nd</sup> step: Establish a high level model that will consolidate all the characteristics of coupling
- 3<sup>rd</sup> step: Gain consensus on this coupling model
- 4<sup>th</sup> step: Integrate the coupling measures into this high level model

***Thank You !***



***Laila.cheikhi.1@ens.etsmtl.ca, aabran@ele.etsmtl.ca, mlo@Cetic.bc***

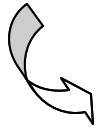


# *Appendix*



# Coupling Summary

- *Different Definitions of coupling*



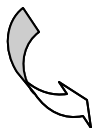
***Lack of Consensus on the concept***

- *Different Models of coupling*



***Lack of High level Model***

- *Different Measures of coupling*



***Different designs for measures***



# Numerical Assignment rules CBO

The class to be measured      The number of classes in the system      The other class : j varies from 1 to NCS      Excludes the measured class when counting

$$CBO(C_i) = \sum_{j=1}^{NCS} [C_j \neq C_i | (m_i \in \{M_i\} \text{ use } ((m_j \in \{M_j\}) \vee (a_j \in \{A_j\}))) \vee (m_j \in \{M_j\} \text{ use } ((m_i \in \{M_i\}) \vee (a_i \in \{A_i\}))) ]$$

The set of all Ci methods      The set of all Ci attributes      The set of all Cj methods      The set of all Cj attributes





# Example of CBO

We take as example thee Classes A, B and C :

```
class A {  
    B b;  
    void mA () { b.mB();}  
}
```

```
class B{  
    int c;  
    void mB() {c= mC();}  
}
```

```
class C {  
    .....  
    void mC() {...}  
}
```

Calculation of the CBO:

- $CBO(A) = 1$ . Method  $mA()$  of class A uses method  $mB()$  of class B.
- $CBO(B) = 2$ . Method  $mB()$  of class B uses method  $mC()$  of class C, and method  $mB()$  is used by method  $mA()$  of class A.
- $CBO(C) = 1$ . Method  $mC()$  of class C is used by method  $mB()$  of class B.

**CBO (X) correspond to the number of classes used by the class X or using the class X**