ECOOP'98
Brussels, July 1998

Reengineering the Modularity of Object Oriented Systems

Fernando Brito e Abreu, ISEG/UTL and INESC
Gonçalo Pereira, INESC

Abstract

Validation experiment of a quantitative approach to the modularization of object oriented systems.

Based on Cluster Analysis, a statistical technique used in many fields of science to group items.

A sample of some relatively large OO systems was used in this experiment.

The similarity between classes was computed on the basis of their relative couplings and on six different rating options.

These couplings are classified according to a taxonomy framework where categories were assigned weights.

The MOOD Project

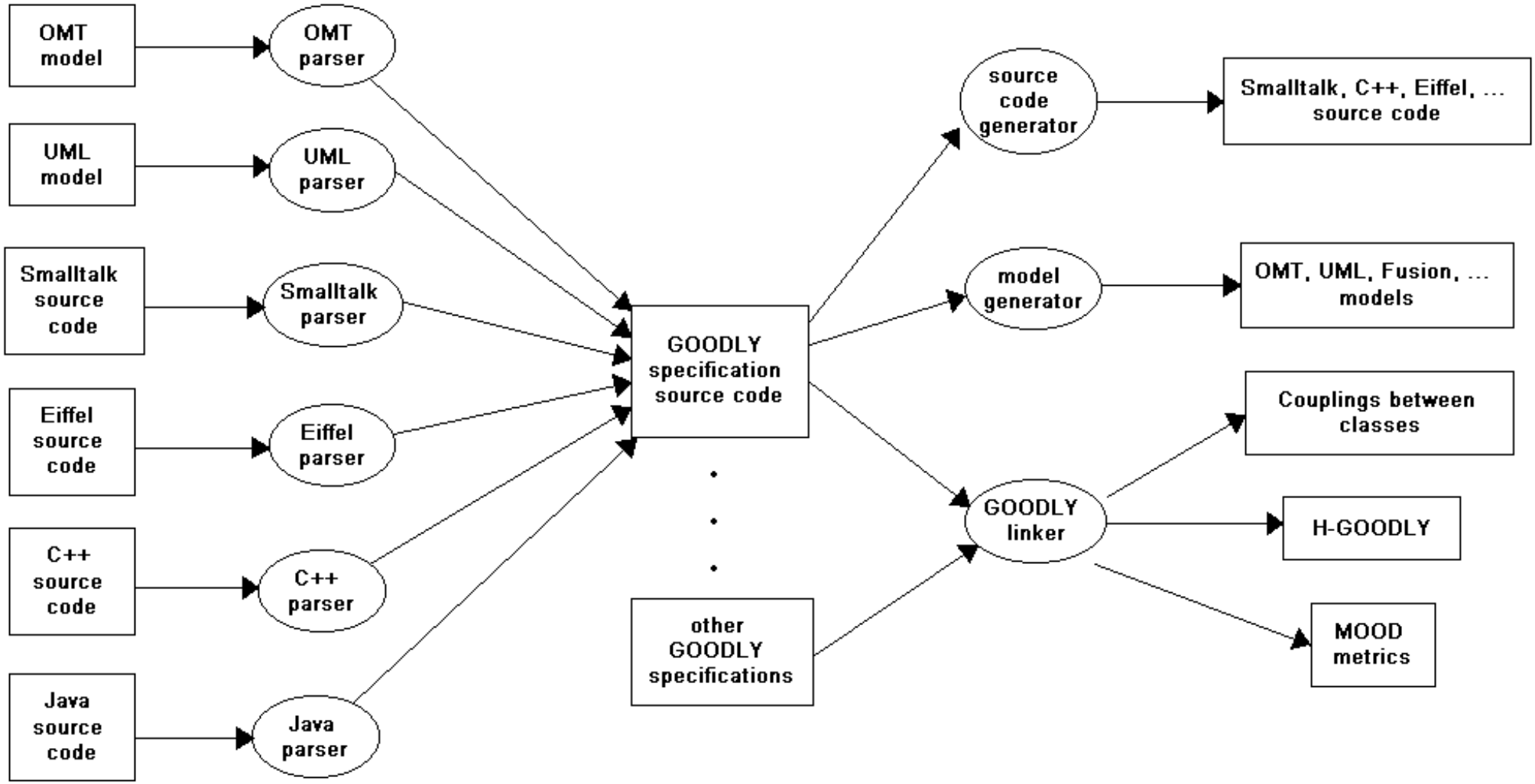
Where: Software Engineering Group at INESC

Main objective: broad spectrum quantitative framework to support the design of object oriented systems.

Aspects covered:

- Automatic OO metrics collection by forward and reverse engineering
- Inheritance and Polymorphism analysis
- Information hiding, Coupling and cohesion analysis
- Modularity and Reuse analysis
- Design heuristics
- Reliability estimation
- Testing effort estimation
- Maintenance effort estimation

MOODKit G2 Architecture



Coupling categories

Acronym	Coupling category
DI	Direct Inheritance
CP	Class Parameter
AT	Attribute Type
EA	Employed Attribute
ET	Employed Type
PO	Parameter in Operation
PM	Parameter in Message
PC	Parameter in Call
RO	Return in Operation
RM	Return in Message
RC	Return in Call
LA	Local Attribute in Operation
MR	Message Recipient

Cluster Analysis

Group of techniques concerned with the classification of similar items into groups in such a way that the relations between items in the same group are stronger than the relations to items in other groups.

We need two things:

- Some measure of the way each item relates to the others and a method to group the items (dissimilarities or distance) - summarized in a dissimilarities matrix.

- An algorithm to drive the clustering process, a clustering method.

CLUSTERING METHODS

Acronym	Method Name
WG	Average Linkage (Within Group) */ Nearest neighbor method*
BG	Average Linkage (Between Groups) */ Furthest neighbor method*
SL	Single Linkage
CL	Complete Linkage
CM	Centroid Method
MM	Median Method
WM	Ward Method

The methodological approach

Affinity between classes

The more coupling there are between two classes, the stronger is their interconnection strength, which we called **affinity (** $A \in [0,\infty[$)

Several schemes of combination of the available coupling information.

	Unweighted	**Weighted**
Binary	$A_{UB}(i,j) = \sum_{cc=DI}^{MR} C_{cc}(i,j)$	$A_{WB}(i,j) = \sum_{cc=DI}^{MR} \alpha_{cc} C_{cc}(i,j)$
Additive	$A_{UA}(i,j) = \sum_{cc=DI}^{MR} N_{cc}(i,j)$	$A_{WA}(i,j) = \sum_{cc=DI}^{MR} \alpha_{cc} N_{cc}(i,j)$
Multiplicative	$A_{UM}(i,j) = \prod_{cc=DI}^{MR} N_{cc}(i,j)$	$A_{WM}(i,j) = \prod_{cc=DI}^{MR} \alpha_{cc} N_{cc}(i,j)$

Similarity and Dissimilarity

The **dissimilarity** between two classes *i* and *j*, depends on the **affinity**, to which was applied a **standardization transformation** so that the dissimilarity D(i,j)∈ [0,1].

$$D(i,j) = \begin{cases} \dfrac{1}{1+A(i,j)} & \forall i \neq j \\ 0 & \forall i = j \end{cases}$$

- For each system under study, six dissimilarities matrixes were generated, one for each affinity rating scheme.
- Those matrixes were fed to SPSS.
- The execution of the clustering algorithm was stopped when the number of groups was the same than the original number of modules in the system.

The sample

Systems selection criteria:

- Diversity, Considerable size, Modularized

Name	Typ	Form.	Clas	Mod	Memb	Coupl	C/Cl	WG/WB
Ems	App	Eiffel	111	8	13,9	4836	5,2	0,5156
ISE Eiffel	Env	Eiffel	150	3	50,0	20626	6,9	0,5314
SIG Eiffel	Env	Eiffel	79	4	19,8	10630	8,2	0,6121
Fec	App	Eiffel	171	6	28,5	21437	13,3	0,6713
Gobo	Lib	Eiffel	15	5	3,0	6722	39,0	0,7442
Stix	App	Smtalk	110	52	2,1	5918	7,0	0,8622
Bast	App	Smtalk	38	11	3,5	1219	4,5	0,8834
Gpackage	Lib	OMT	46	18	2,6	167	2,1	0,9333
GNU SmallTalk	Env	Smtalk	246	104	2,4	10040	6,2	0,9469

Clustering proposals (example)

Classes	WG	BG	SL	CL	CM	MM	WM
GPADrawingArea	1	1	1	1	1	1	1
GPDrawingStyle	1	1	2	2	1	1	1
GPApplication	2	2	3	3	2	2	2
GPWindow	3	3	4	4	3	3	3
GPLabel	4	4	5	5	4	1	4
GPManager	5	5	6	6	1	1	5
GPButton	4	4	7	7	1	1	4
GPPushButton	4	4	8	8	5	1	4
GPFramedPie	10	10	1	1	1	1	10
GPArc	10	10	1	14	1	1	10
GPSlider	17	17	17	1	1	1	17
GPRealSlider	17	17	17	1	17	1	17
GPIntSlider	17	17	17	1	1	1	17
GPTextArea	18	18	18	1	18	18	18

Generation of inclusion matrixes

	A	B	C	D	E	F	G	H	I	J
A	1	1	0	0	0	0	0	0	0	0
B	1	1	0	0	0	0	0	0	0	0
C	0	0	1	1	1	0	0	0	0	0
D	0	0	1	1	1	0	0	0	0	0
E	0	0	1	1	1	0	0	0	0	0
F	0	0	0	0	0	1	1	1	1	1
G	0	0	0	0	0	1	1	1	1	1
H	0	0	0	0	0	1	1	1	1	1
I	0	0	0	0	0	1	1	1	1	1
J	0	0	0	0	0	1	1	1	1	1

Matching algorithm between original and generated inclusion matrixes

$$M = \frac{sucessful\ maches}{\left(N^2 - N\right)/2}$$

	WG	BG	SL	CL	CM	MM	WM
UB	0.9433	0.9182	0.7820	0.8115	0.6276	0.6370	0.9526
WB	0.9469	0.9318	0.8728	0.8182	0.6274	0.6393	0.9533
UA	0.9371	0.8973	0.6402	0.8213	0.6274	0.6312	0.9565
WA	0.9417	0.9221	0.6710	0.8308	0.6282	0.6357	0.9561
UM	0.9378	0.8999	0.6395	0.8210	0.6272	0.6315	0.9549
WM	0.9371	0.9207	0.6826	0.8336	0.6276	0.6432	0.9548

"GNU SmallTalk" matchings

	WG	BG	SL	CL	CM	MM	WM
UB	0.9372	0.9169	0.8734	0.8000	0.6802	0.5768	0.9333
WB	0.9333	0.9449	0.8725	0.8000	0.7314	0.6184	0.9314
UA	0.9275	0.9643	0.6976	0.8213	0.5961	0.5556	0.9391
WA	0.9382	0.9498	0.8618	0.8386	0.7372	0.5836	0.9353
UM	0.9266	0.9111	0.6580	0.8077	0.6174	0.5633	0.9362
WM	0.9382	0.9498	0.8618	0.8203	0.7101	0.5855	0.9353

"GPackage" matchings

	WG	BG	SL	CL	CM	MM	WM
UB	0.8414	0.8359	0.9156	0.9014	0.6110	0.6214	0.8824
WB	0.8622	0.8432	0.9354	0.9016	0.6083	0.6087	0.8866
UA	0.8407	0.7099	0.7091	0.9141	0.6160	0.6100	0.8782
WA	0.8457	0.7743	0.7091	0.8969	0.6133	0.6177	0.8817
UM	0.8450	0.7648	0.7178	0.9179	0.6160	0.6100	0.8782
WM	0.8462	0.7850	0.7291	0.9026	0.6137	0.6212	0.8816

"Stix" matchings

Matching Conclusions

Weak impact

- affinity ratings
- number of couplings, either in absolute or relative value (average per class).

Strong impact of clustering methods

- two best: WG and WM
- two worst: MM and CM.

Strong impact:

Average number of classes per module (first classes to be clustered are the ones who show the greatest similarity!)

Corollary

For systems whose average module membership (number of classes per module) is not too high, the proposed cluster analysis approach leads to modularization solutions that **match very closely those proposed by human experts**.

The approach presented can be used in the **reengineering of object oriented legacy systems**, allowing to **identify ill-defined modularization** situations and to **propose alternative ones**.