

# Relative Thresholds: Case Study to Incorporate Metrics in the Detection of Bad Smells

Authors:



[www.giro.infor.uva.es](http://www.giro.infor.uva.es)

---

Yania Crespo González-Carvajal  
Carlos López Nozal  
Raúl Marticorena Sánchez

[yania@infor.uva.es](mailto:yania@infor.uva.es)  
[clopezno@ubu.es](mailto:clopezno@ubu.es)  
[rmartico@ubu.es](mailto:rmartico@ubu.es)

# Outline

- 2 ● Introduction
  - Context
  - State of the Art
  - Previous Works
  - Outlined Problem
- Case Study
  - Phase 1: Comparison between products
    - Partial conclusions
  - Phase 2: Metric evolution between versions
    - Partial conclusions
- Applying Relative Product Thresholds
  - Example
- Conclusions
- Proposal and Future Work

## Context

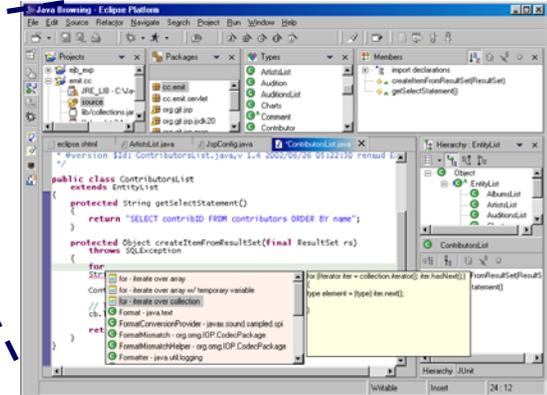
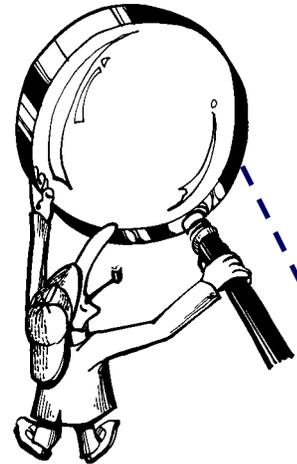
3

### ■ When should we refactor?

- Visual inspection
  - Too much time!

### ■ Metric collection tools

- Automatic process
- Using these values in smell detection
  - Thresholds helping to point out their presence
- Some research works but not included in most of the current IDEs



## Context Current solutions in IDEs

### 4 ■ Example: Eclipse 3.1 & Metrics 1.3.6 plug-in

Here you can set the safe range for each metric.  
Metric values outside these ranges result in warnings if warnings are enabled.

Metric	Level	Min	Max	Hint for fix
Number of Static Methods	type			
Total Lines of Code	compilationUnit			
Afferent Coupling	packageFragment			
Normalized Distance	packageFragment			
Number of Classes	compilationUnit			
Specialization Index	type			
Instability	packageFragment			
Number of Attributes	type			
Number of Packages	packageFragmentRoot			
Method Lines of Code	method			
Weighted methods per Class	type			
Number of Overridden Methods	type			
Number of Static Attributes	type			
Nested Block Depth	method		5.0	use Extract-method to split the method up
Number of Methods	type			
Lack of Cohesion of Methods	type			
McCabe Cyclomatic Complexity	method		10.0	use Extract-method to split the method up
Number of Parameters	method		5.0	Move invoked method or pass an object
Abstractness	packageFragment			
Number of Interfaces	compilationUnit			
Efferent Coupling	packageFragment			
Number of Children	type			
Depth of Inheritance Tree	type			

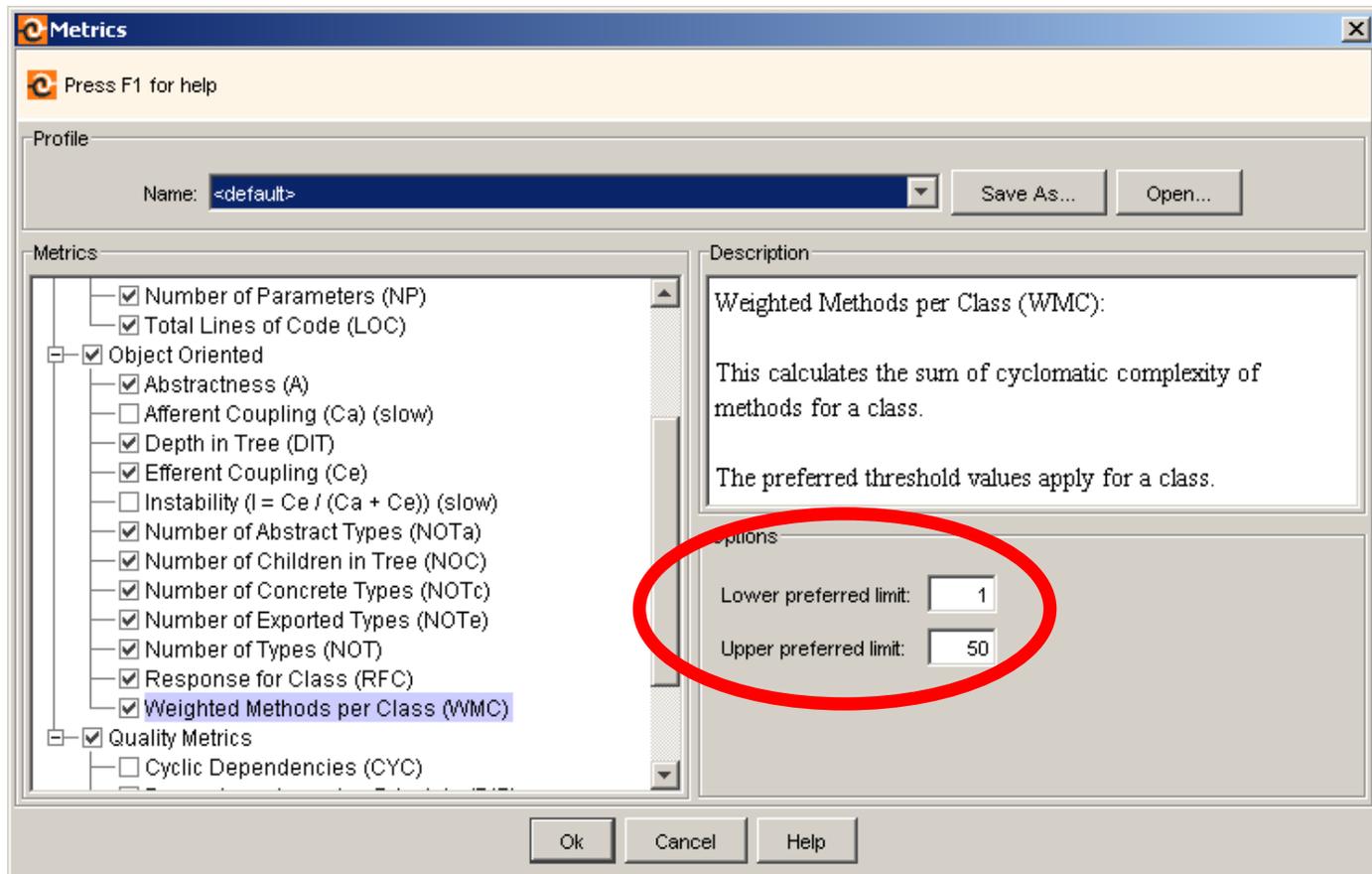
Restore Defaults Apply

OK Cancel

## Context Current solutions in IDEs

5

### ■ Example: RefactorIt



July, 3rd, 2006

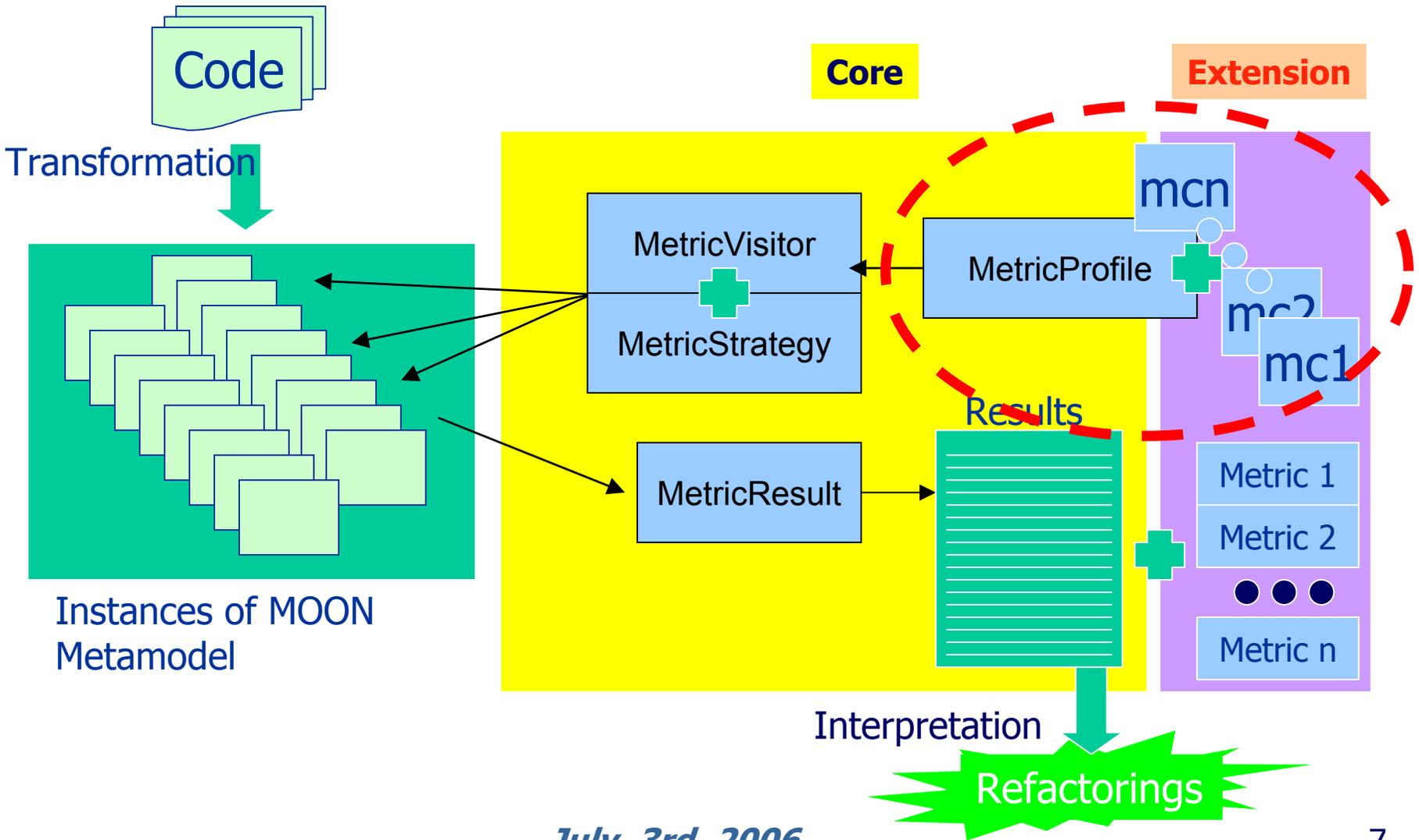
## State of the Art

6

- **Establishing Software Metrics** [French, 1999]
  - Method to define relative thresholds
  - Neither working in refactoring nor bad smells
- **Bad Smells Taxonomy** [Mantyla, 2003]
  - Metrics suggest bad smells
  - Absolute values
- **Flaw Detection** [Marinescu, 2002]
  - Using metrics
  - Absolute and relative filters
- **SOUL** [Muñoz, 2003]
  - Logic predicates: metrics and heuristics
  - Generic thresholds

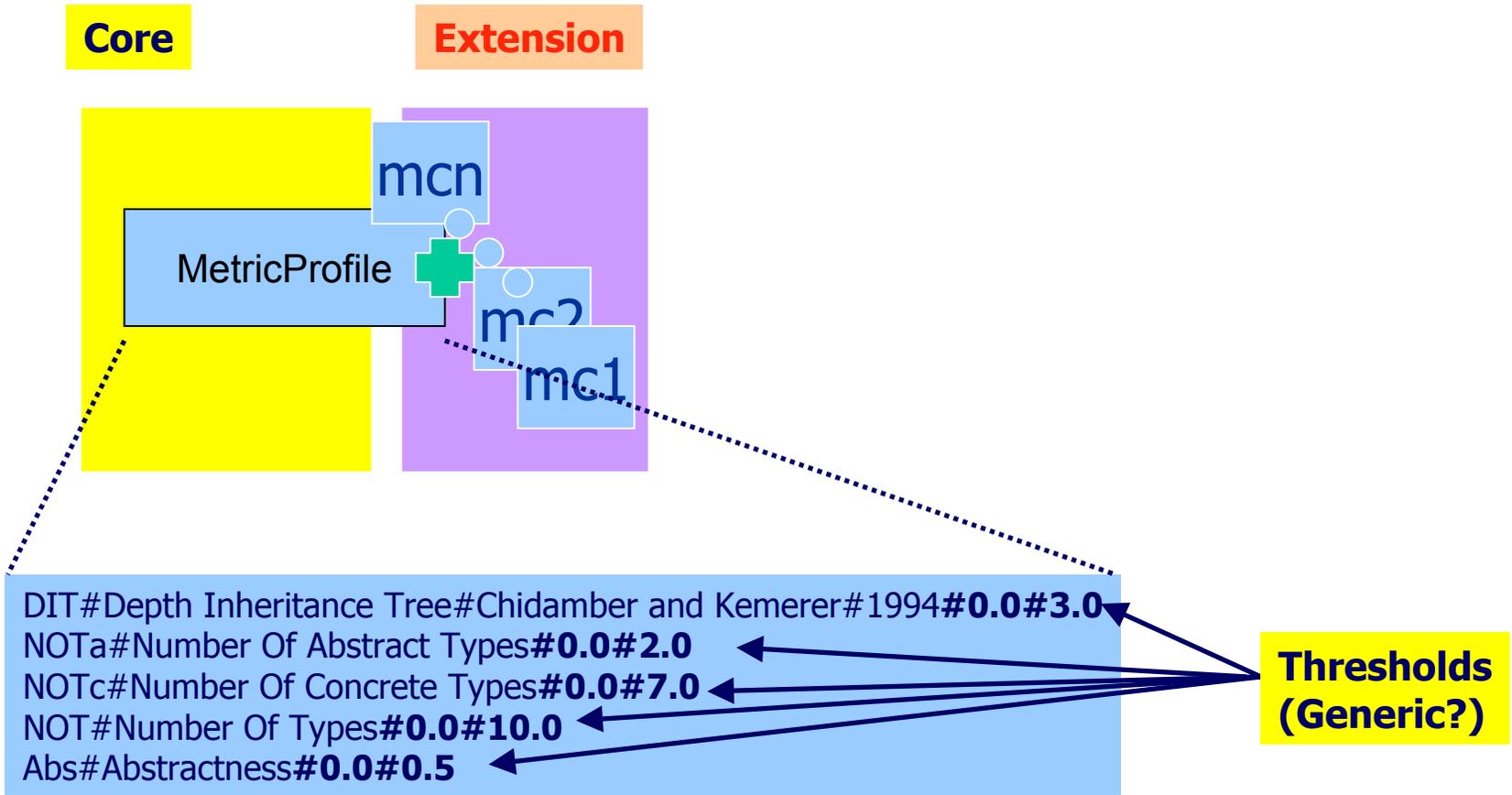
# Previous Work Support Based on Frameworks [QA00SE'05]

7



# Previous Work Support Based on Frameworks [QA00SE'05]

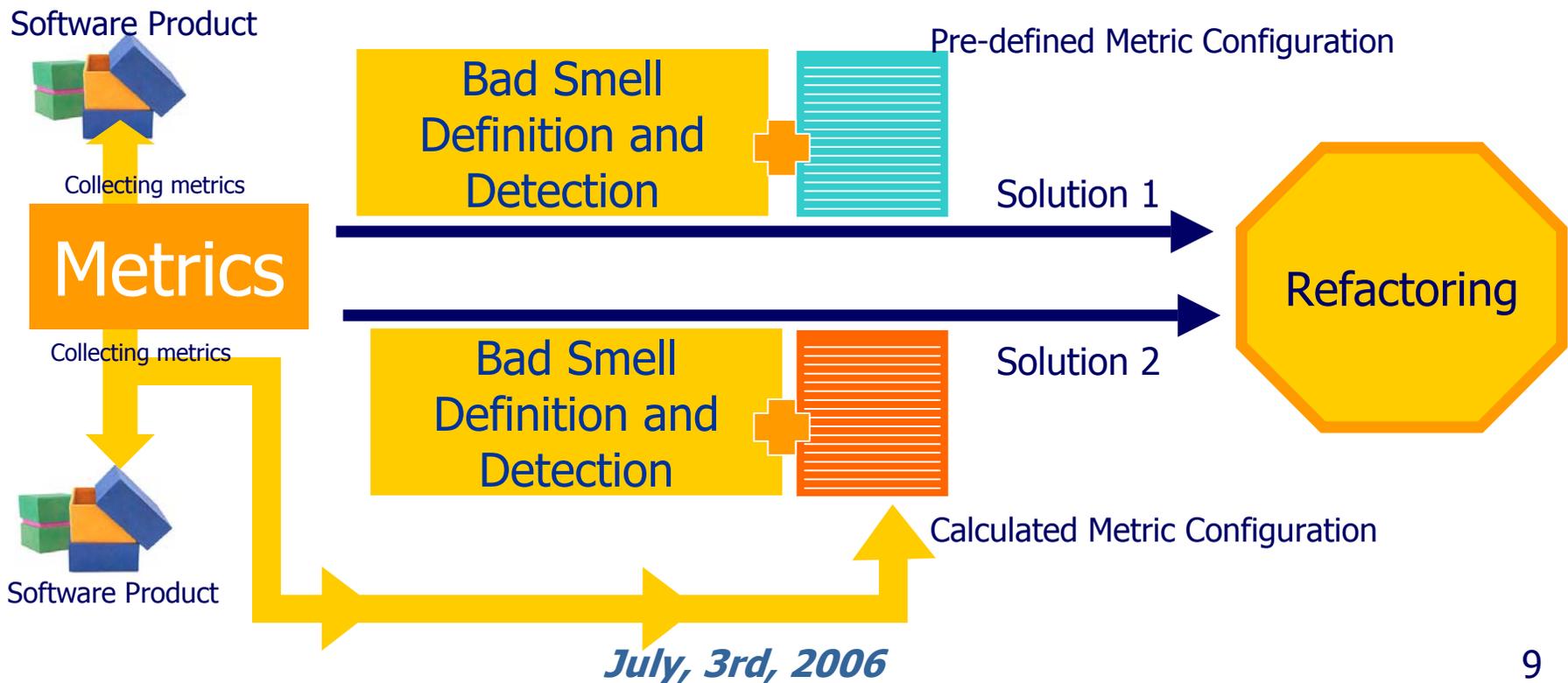
8





# Outlined Problem

- Should we use generic or relative product thresholds to detect bad smells?
- And then... refactor the code



## Case Study

### Phase 1: Comparison between products

10

- Phase 1: Comparison between products
  - Six software products – stable - medium size
    - jfreechart-1.0.0.pre2 (629 classes)
    - jhotdraw-6.0b1 (496 classes)
    - struts-1.2.8 (273 classes)
    - jcoverage-1.0.5 (90 classes)
    - easymock-1.0.5 (47 classes)
    - junit-3.8.1 (46 classes)
  - Collecting metrics over different features:
    - NOF number of fields
    - NOM number of methods
    - WMC cyclomatic complexity
    - LCOM lack of cohesion of methods
    - DIT depth in the inheritance tree
    - NSC number of children
    - SIX specialization index
    - NORM number of overridden methods

## Case Study

### Phase 1: Comparison between products

11

## Results

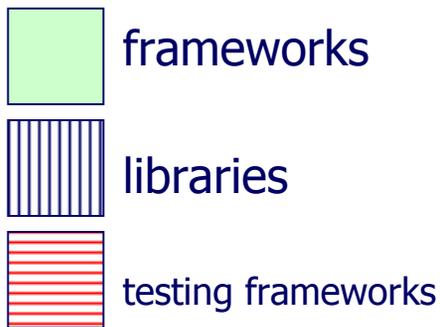
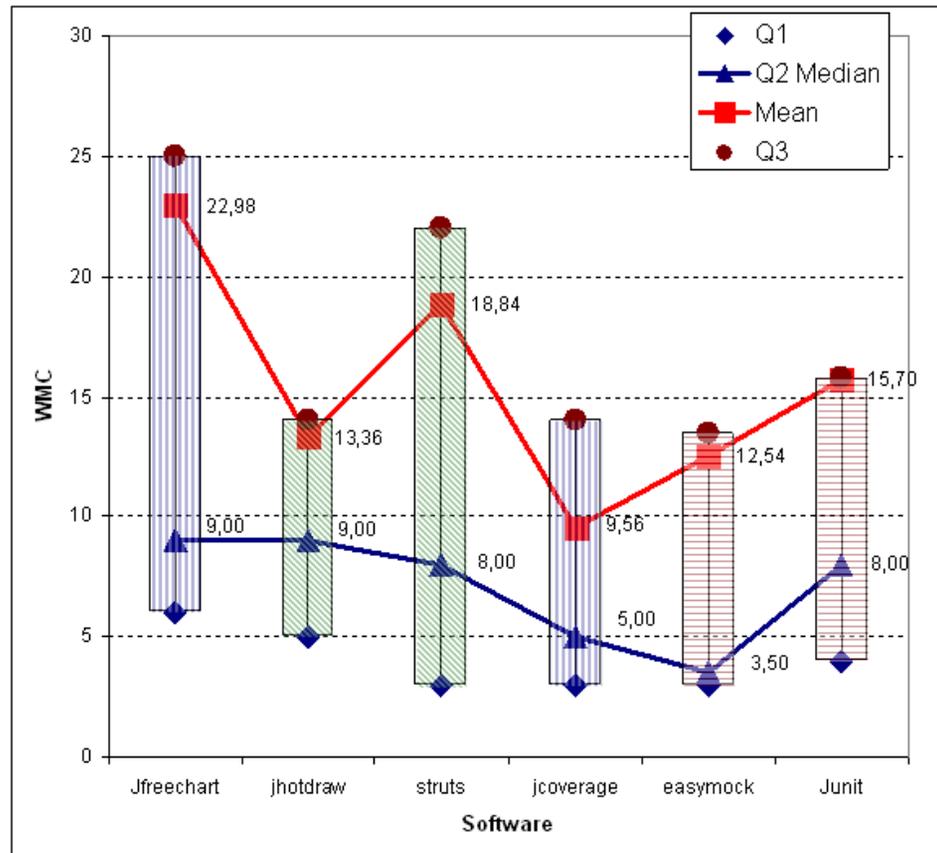
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JFreeChart 1.0.0-pre2	2,40	10,08	22,98	0,21	2,55	0,36	0,16	0,69
Bounded mean (15%)	1,41	7,45	15,87	0,17	2,47	0,04	0,08	0,46
Q3	<b>3,00</b>	<b>11,00</b>	<b>25,00</b>	0,50	3,00	0,00	0,14	1,00
Q2 Median	1,00	5,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	<b>0,00</b>	<b>3,00</b>	<b>6,00</b>	0,00	2,00	0,00	0,00	0,00
Standard Deviation	5,05	15,01	38,82	0,32	1,14	1,48	0,37	1,23
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	48,00	166,00	490,00	1,00	7,00	16,00	3,20	9,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean Junt-3.8,1	2,17	8,13	15,70	0,21	2,70	0,28	0,18	0,35
Bounded mean (15%)	1,50	6,53	12,33	0,18	2,58	0,15	0,09	0,28
Q3	<b>2,00</b>	<b>9,75</b>	<b>15,75</b>	0,50	3,75	0,00	0,12	1,00
Q2 Median	1,00	4,50	8,00	0,00	2,00	0,00	0,00	0,00
Q1	<b>0,00</b>	<b>2,00</b>	<b>4,00</b>	0,00	1,00	0,00	0,00	0,00
Standard Deviation	3,59	10,35	20,42	0,33	1,84	0,72	0,45	0,60
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	18,00	62,00	106,00	0,91	6,00	3,00	2,00	3,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean Jcoverage-1.0.5	1,49	4,35	9,56	0,24	1,78	0,39	0,81	0,28
Bounded mean (15%)	1,23	3,70	8,17	0,20	1,62	0,19	0,10	0,21
Q3	<b>2,00</b>	<b>5,00</b>	<b>14,00</b>	0,50	2,00	0,00	0,00	0,00
Q2 Median	1,00	3,00	5,00	0,00	1,00	0,00	0,00	0,00
Q1	<b>0,00</b>	<b>2,00</b>	<b>3,00</b>	0,00	1,00	0,00	0,00	0,00
Standard Deviation	1,87	4,46	9,59	0,34	1,05	0,96	0,37	0,52
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	7,00	25,00	46,00	1,00	5,00	4,00	1,67	2,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean easymock-2.0	1,41	5,83	12,54	0,15	1,24	0,09	0,12	0,33
Bounded mean (15%)	1,24	4,13	8,32	0,11	1,08	0,00	0,02	0,16
Q3	<b>2,00</b>	<b>5,00</b>	<b>13,50</b>	0,33	1,00	0,00	0,00	0,00
Q2 Median	1,00	3,00	3,50	0,00	1,00	0,00	0,00	0,00
Q1	<b>1,00</b>	<b>3,00</b>	<b>3,00</b>	0,00	1,00	0,00	0,00	0,00
Standard Deviation	1,34	7,51	19,25	0,24	0,67	0,46	0,41	0,73
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	6,00	38,00	105,00	0,85	4,00	3,00	2,00	3,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean struts-1.2.8	2,91	8,60	18,84	0,28	2,59	0,46	0,51	0,96
Bounded mean (15%)	2,09	6,66	13,21	0,25	2,45	0,24	0,33	0,67
Q3	<b>4,00</b>	<b>11,00</b>	<b>22,00</b>	0,67	4,00	1,00	0,60	1,00
Q2 Median	2,00	4,00	8,00	0,00	2,00	0,00	0,00	0,00
Q1	<b>0,00</b>	<b>2,00</b>	<b>3,00</b>	0,00	1,00	0,00	0,00	0,00
Standard Deviation	4,56	11,02	29,13	0,36	1,48	1,13	0,95	2,04
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	40,00	82,00	260,00	0,98	7,00	10,00	5,00	28,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JHotDraw60b1	1,40	9,51	13,36	0,16	2,84	0,57	0,31	0,73
Bounded mean (15%)	1,09	7,72	10,31	0,11	2,68	0,07	0,16	0,38
Q3	<b>2,00</b>	<b>11,00</b>	<b>14,00</b>	0,00	4,00	0,00	0,32	1,00
Q2 Median	1,00	7,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	<b>0,00</b>	<b>4,00</b>	<b>5,00</b>	0,00	2,00	0,00	0,00	0,00
Standard Deviation	1,86	10,40	16,76	0,30	1,49	3,84	0,74	1,70
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	19,00	90,00	158,00	1,50	9,00	71,00	8,00	19,00

## Partial Conclusions

Phase 1: Comparison between products

### 12 ■ Example: WMC

- box plot removing "whiskers"
- different values between distributions



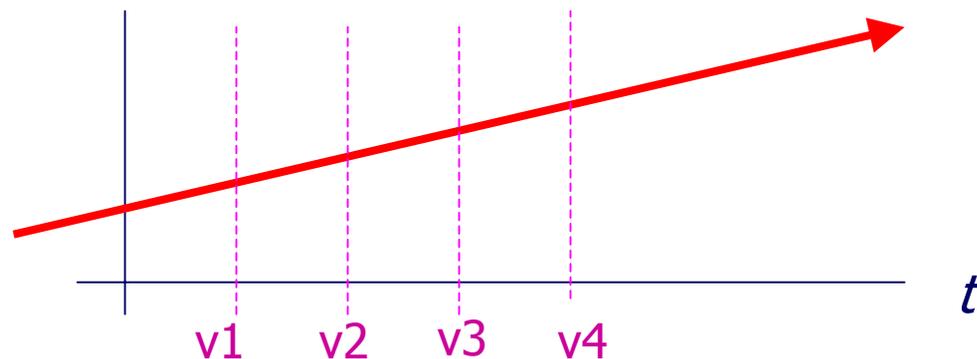
July, 3rd, 2006

## Partial Conclusions

### Phase 1: Comparison between products

13

- Initial observations and conclusions
  - Distributions are not symmetrical with positive asymmetry
  - Certain correlation of some metrics with size product
  - Dispersed data
    - Differences between minimum and maximum values are large
    - Very different between products
  - Kind of product does not determine distribution size
- Question
  - Does absence of generic thresholds generate large metric changes between versions?



*July, 3rd, 2006*

## Case Study

### Phase 2: Metric evolution between versions

14

#### ■ Phase 2: Metric evolution between versions

- Taking different versions of three products collecting the same metrics
  - along several years, even duplicating the number of classes from initial versions



- `jfreechart-1.0.1` (691 classes, 2006-01-27)
- `jfreechart-1.0.0-pre2` (629 classes, 2005-03-10)
- `jfreechart-0.9.21` (570 classes, 2004-09-10)
- `jfreechart-0.9.7` (492 classes, 2003-04-17)
- `jfreechart-0.9.4` (326 classes, 2002-10-18)

#### **JHotDraw**

- `jhotdraw-6.0b1` (497 classes, 2004-02-01)
- `jhotdraw-5.4b2` (478 classes, 2004-01-31)
- `jhotdraw-5.3` (208 classes, 2002-01-20)
- `jhotdraw-5.2` (149 classes, 2001-02-18)



- `junit-3.8.1` (47 classes)
- `junit-3.2` (32 classes)
- `junit-2.1` (19 classes)

*July, 3rd, 2006*

## Case Study

### Phase 2: Metric evolution between versions

15

- Results
- JFreeChart

	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jfreechart-1.0.1	2,22	9,94	22,42	0,19	2,53	0,33	0,16	0,69
Bounded mean (15%)	1,27	7,27	15,25	0,15	2,46	0,03	0,09	0,48
Q3	2,00	11,00	23,00	0,40	3,00	0,00	0,17	1,00
Q2 Median	1,00	5,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	4,00	7,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	4,86	15,18	39,39	0,31	1,12	1,41	0,35	1,18
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	46,00	173,00	513,00	1,00	7,00	14,00	3,33	8,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jfreeChart-1.0.0-pre2	2,40	10,08	22,98	0,21	2,55	0,36	0,16	0,69
Bounded mean (15%)	1,41	7,45	15,87	0,17	2,47	0,04	0,08	0,46
Q3	3,00	11,00	25,00	0,50	3,00	0,00	0,14	1,00
Q2 Median	1,00	5,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	3,00	6,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	5,05	15,01	38,82	0,32	1,14	1,48	0,37	1,23
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	48,00	166,00	490,00	1,00	7,00	16,00	3,20	9,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jfreechart-0.9.21	2,38	9,99	22,47	0,21	2,52	0,36	0,16	0,66
Bounded mean (15%)	1,41	7,33	15,44	0,17	2,45	0,05	0,08	0,44
Q3	2,00	12,75	26,00	0,50	3,00	0,00	0,16	1,00
Q2 Median	1,00	5,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	3,00	6,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	4,93	15,20	38,66	0,32	1,12	1,47	0,37	1,20
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	47,00	155,00	473,00	0,96	7,00	16,00	3,00	8,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jfreechart-0.9.7	2,14	7,03	15,63	0,20	3,21	0,31	0,17	0,49
Bounded mean (15%)	1,22	5,06	11,08	0,15	3,07	0,04	0,07	0,28
Q3	2,00	9,00	19,00	0,50	4,00	0,00	0,09	1,00
Q2 Median	0,00	3,00	6,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	1,00	3,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	4,55	10,65	24,45	0,32	1,97	1,34	0,44	1,02
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	39,00	87,00	203,00	1,00	7,00	15,00	3,00	7,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jfreechart-0.9.4	2,69	7,77	18,00	0,26	3,02	0,40	0,27	0,61
Bounded mean (15%)	1,71	6,13	13,51	0,22	2,85	0,08	0,11	0,32
Q3	3,00	11,00	24,00	0,62	4,00	0,00	0,16	1,00
Q2 Median	1,00	4,00	8,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	1,00	3,00	0,00	1,00	0,00	0,00	0,00
Standard Deviation	4,87	9,70	25,11	0,35	1,95	1,49	0,70	1,34
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	39,00	60,00	195,00	1,00	7,00	16,00	6,00	8,00

## Case Study

### Phase 2: Metric evolution between versions

16

## Results

### JHotDraw

	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JHotDraw60b1	1,40	9,51	13,36	0,16	2,84	0,57	0,31	0,73
Bounded mean (15%)	1,09	7,72	10,31	0,11	2,68	0,07	0,16	0,38
Q3	2,00	11,00	14,00	0,00	4,00	0,00	0,32	1,00
Q2 Median	1,00	7,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	0,00	4,00	5,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	1,86	10,40	16,76	0,30	1,49	3,84	0,74	1,70
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	19,00	90,00	158,00	1,50	9,00	71,00	8,00	19,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jhotdraw54b1	1,41	9,67	13,89	0,16	2,90	0,58	0,32	0,73
Bounded mean (15%)	1,12	7,85	10,80	0,11	2,75	0,08	0,17	0,39
Q3	2,00	11,00	15,00	0,00	4,00	0,00	0,33	1,00
Q2 Median	1,00	7,00	9,00	0,00	3,00	0,00	0,00	0,00
Q1	1,00	4,00	6,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	1,81	10,33	16,88	0,30	1,48	3,89	0,75	1,71
Minimum	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
Maximum	16,00	88,00	148,00	1,50	9,00	71,00	8,00	19,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jhotdraw53	1,83	9,12	15,51	0,27	2,65	0,86	0,51	1,21
Bounded mean (15%)	1,46	7,07	11,60	0,23	2,43	0,20	0,41	0,97
Q3	3,00	10,00	18,00	0,63	3,00	0,00	0,75	2,00
Q2 Median	1,50	6,50	12,50	0,00	1,00	0,00	0,00	0,00
Q1	0,00	3,00	4,75	0,00	2,00	0,00	0,00	0,00
Standard Deviation	2,38	10,87	20,54	0,35	1,66	3,53	0,66	1,66
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	17,00	72,00	146,00	1,50	8,00	40,00	3,00	12,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean jhotdraw52	1,83	8,30	13,53	0,26	2,81	0,68	0,56	1,28
Bounded mean (15%)	1,52	6,37	10,25	0,23	2,60	0,24	0,48	1,06
Q3	3,00	10,00	15,25	0,60	3,00	0,00	1,00	2,00
Q2 Median	1,00	5,00	8,00	0,00	2,00	0,00	0,28	1,00
Q1	0,00	3,00	4,00	0,00	2,00	0,00	0,00	0,00
Standard Deviation	2,19	9,82	16,84	0,33	1,70	1,91	0,66	1,69
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	14,00	61,00	108,00	1,50	8,00	12,00	3,11	12,00

## Case Study

### Phase 2: Metric evolution between versions

17

- Results
- JUnit

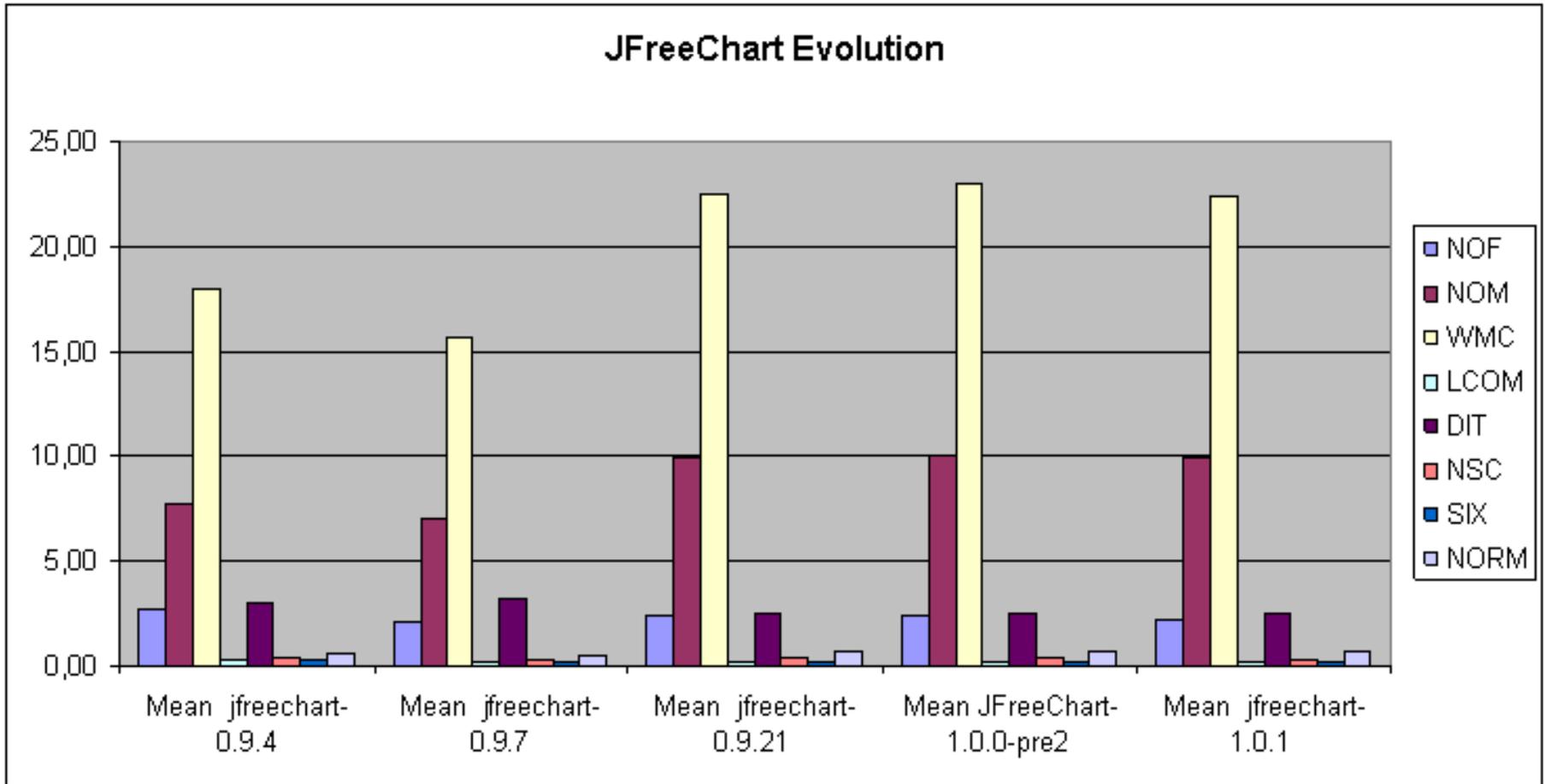
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JUnit 3.8.1	2,17	8,13	15,70	0,21	2,70	0,28	0,18	0,35
Bounded mean (15%)	1,50	6,53	12,33	0,18	2,58	0,15	0,09	0,28
Q3	2,00	9,75	15,75	0,50	3,75	0,00	0,12	1,00
Q2 Median	1,00	4,50	8,00	0,00	2,00	0,00	0,00	0,00
Q1	0,00	2,00	4,00	0,00	1,00	0,00	0,00	0,00
Standard Deviation	3,59	10,35	20,42	0,33	1,84	0,72	0,45	0,60
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	18,00	62,00	106,00	0,91	6,00	3,00	2,00	3,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JUnit 3.2	2,72	7,94	14,75	0,25	2,56	0,19	0,13	0,34
Bounded mean (15%)	1,68	5,96	11,29	0,22	2,43	0,04	0,09	0,25
Q3	3,00	11,00	18,50	0,50	3,50	0,00	0,19	1,00
Q2 Median	1,00	3,50	5,50	0,00	2,00	0,00	0,00	0,00
Q1	0,00	2,00	2,00	0,00	1,00	0,00	0,00	0,00
Standard Deviation	4,85	11,65	21,05	0,34	1,93	0,64	0,24	0,65
Minimum	0,00	0,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	20,00	60,00	103,00	0,92	6,00	3,00	0,75	3,00
	NOF	NOM	WMC	LCOM	DIT	NSC	SIX	NORM
Mean JUnit 2.1	2,16	8,11	14,05	0,22	2,53	0,32	0,31	0,58
Bounded mean (15%)	1,35	7,18	12,41	0,19	2,41	0,18	0,17	0,47
Q3	2,00	8,50	18,00	0,50	3,00	0,00	0,26	1,00
Q2 Median	1,00	4,00	6,00	0,00	2,00	0,00	0,00	0,00
Q1	0,00	4,00	4,00	0,00	1,00	0,00	0,00	0,00
Standard Deviation	4,06	8,52	15,30	0,31	1,61	0,82	0,70	0,84
Minimum	0,00	1,00	1,00	0,00	1,00	0,00	0,00	0,00
Maximum	18,00	31,00	55,00	0,89	6,00	3,00	3,00	3,00

# Case Study

## Phase 2: Metric evolution between versions

18

- Example: Evolution in JFreeChart during 4 years



## Partial Conclusions

Phase 2: Metric evolution between versions

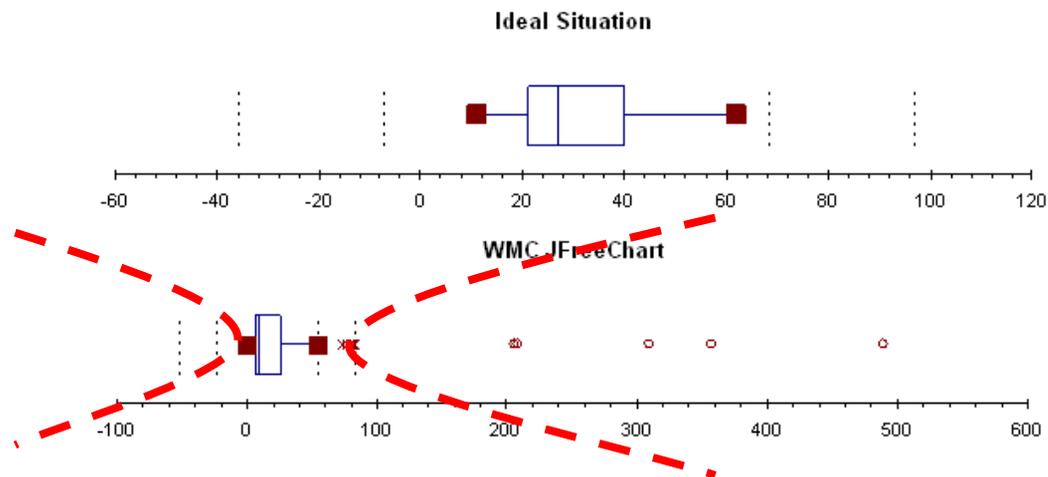
19

- For each product, its relative threshold is stable between versions
- Thresholds should be defined depending on the product
  - can be fixed between stable versions
- Problem: *How can we fix thresholds in first versions?*
  - estimation from similar products (functionality, size, ...)

# Applying Relative Product Thresholds

20

- Many *bad smells* are defined as
  - “high” or “low” values (of metrics)
- Based on previous definition:
  - high and low values could be identified as outliers in distributions



July, 3rd, 2006

# Applying Relative Product Thresholds

21

## ■ Example

### ■ Lazy Class

- *"Classes are not doing enough..."*

- Used criterion:

- $\text{NOF} \leq Q1_{\text{NOF}} \text{ AND } \text{NOM} \leq Q1_{\text{NOM}} \text{ AND } \text{WMC} \leq Q1_{\text{WMC}}$

### ■ Large Class

- *"Classes too large..."*

- Used criterion:

- $\text{NOF} \geq Q3_{\text{NOF}} \text{ AND } \text{NOM} \geq Q3_{\text{NOM}} \text{ AND } \text{WMC} \geq Q3_{\text{WMC}}$

# Applying Relative Product Thresholds

22

- For example:
  - *"What happens if we apply same criteria used with JUnit on JFreechart?"*
  - Number of suspicious classes is very different
- **Lazy Class**
  - 63 suspicious classes
    - $Q1_{NOF}=0$ ,  $Q1_{NOM}=2$ ,  $Q1_{WMC}=4$ , with JUnit values
  - 97 suspicious classes
    - $Q1_{NOF}=0$ ,  $Q1_{NOM}=3$ ,  $Q1_{WMC}=6$ , with JFreeChart new calculated value
- **Large Class**
  - 148 suspicious classes
    - $Q3_{NOF}=2$ ,  $Q3_{NOM}=9.75$ ,  $Q3_{WMC}=15.75$ , with JUnit values
  - 108 suspicious classes
    - $Q3_{NOF}=3$ ,  $Q3_{NOM}=11$ ,  $Q3_{WMC}=25$ , with JFreeChart new calculated values

# Conclusions

23

- Use of generic thresholds with metric collection
- Not the best solution to detect bad smells
- We give an approach using relative thresholds in bad smell detection
  - has been pointed their suitability
  - intended process
    - iteratively calculate metrics, detect bad smells, correct through refactorings, recalculate metrics and check outliers
    - absence (small number) of bad smells stops the process

# Proposal and Future Work

24

- Relative product thresholds should be integrated in our current proposal of bad smell detection
  - Step towards tools that support metric collection and bad smells



- Need of more examples:
  - What happens with other languages?
  - What happens with other kind of products?
  - Does programmer experience, background, culture, ... have any effect?



### Relative Thresholds: Case Study to Incorporate Metrics in the Detection of Bad Smells



Authors:

Yania Crespo González-Carvajal  
Carlos López Nozal  
Raúl Marticorena Sánchez

[yania@infor.uva.es](mailto:yania@infor.uva.es)  
[clopezno@ubu.es](mailto:clopezno@ubu.es)  
[rmartico@ubu.es](mailto:rmartico@ubu.es)

