UFMG

lab-soft
software engineering laboratory

A Method to Derive Metric Thresholds for Software Product Lines

Authors: Gustavo Vale, Eduardo Figueiredo
Presenter: Allan Victor Mori

Software Engineering Lab (LabSoft)
http://labsoft.dcc.ufmg.br/

Agenda

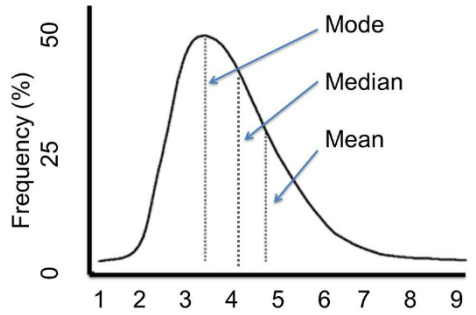
- Concepts
- Objective
- Related Work
- Proposed Method
- Example of Use
- Evaluation
- Threats to Validity
- Conclusion

Concepts

- System Line of Products
  - "Configurable set of systems, shares a common, managed set of features"
  - Reuse of software assets (resources)
- Metrics
  - Access different quality aspects
  - Ex.: Large class realizing non-cohesive operations

Concepts

- Thresholds use
  - Objective characterize or to classify each component



- Methods desire
  - 1) Well defined method
  - 2) Method that consider the skewed distribution
  - 2) Derived from Benchmark

Objective

- Propose a method to derive thresholds that considers the desirable points proposed in previous work.

Related Work

- Software Metric
  - Lines of Code (LOC)
  - Coupling Between Objects (CBO)
  - Weighted Method per Class (WMC)
  - Number of Constant Refinement (NCR)
    - Complexity between Constants and features.

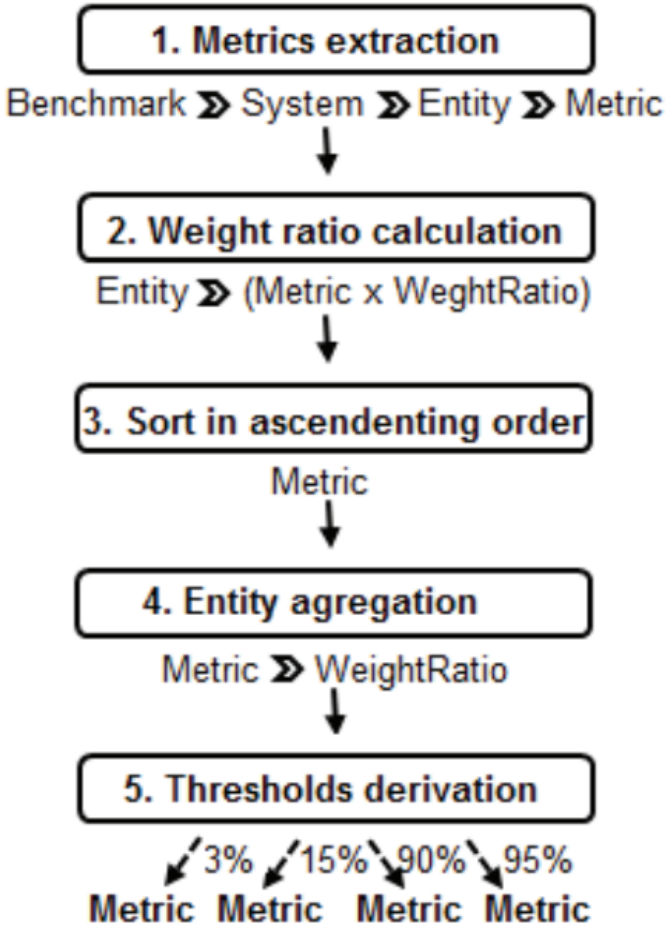
Related Work

- In other works:
  - Thresholds derived from programming experience
  - Thresholds derived form metric analysis

Related Work

- In other works:
  - Methodologies for characterizing metrics distribution
  - Methods to derive thresholds

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

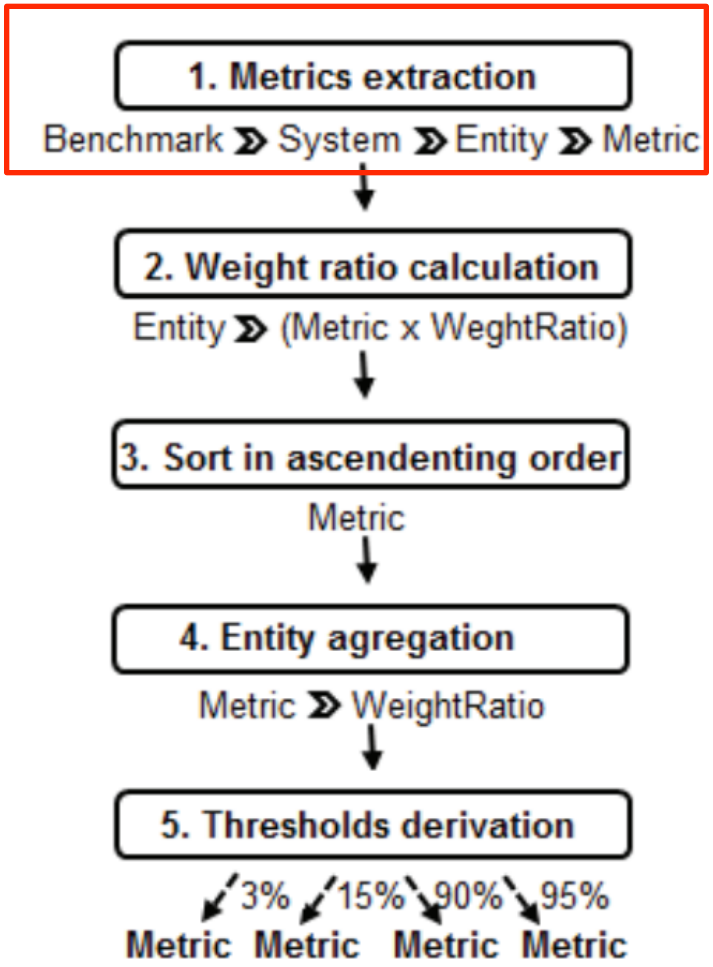
Benchmark
Represents a set of systems

System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

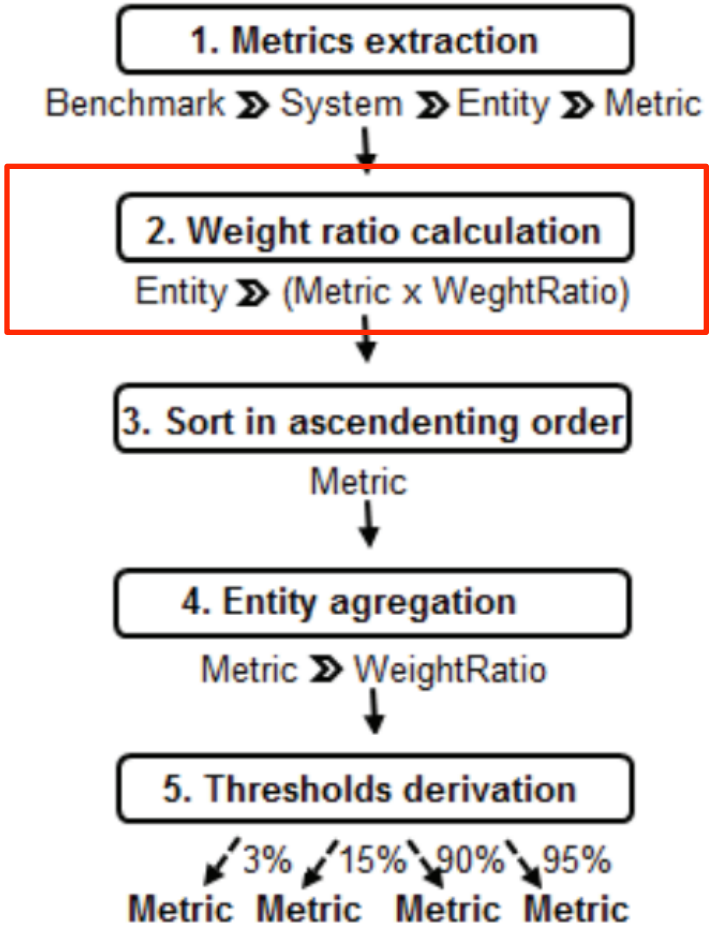
Benchmark
Represents a set of systems

System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

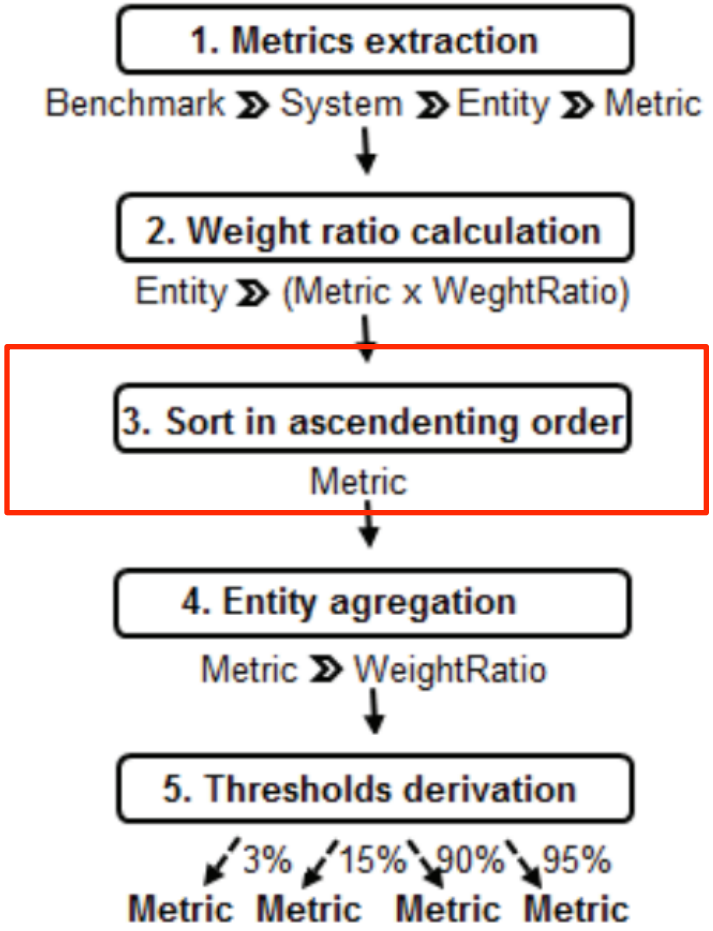
Benchmark
Represents a set of systems

System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

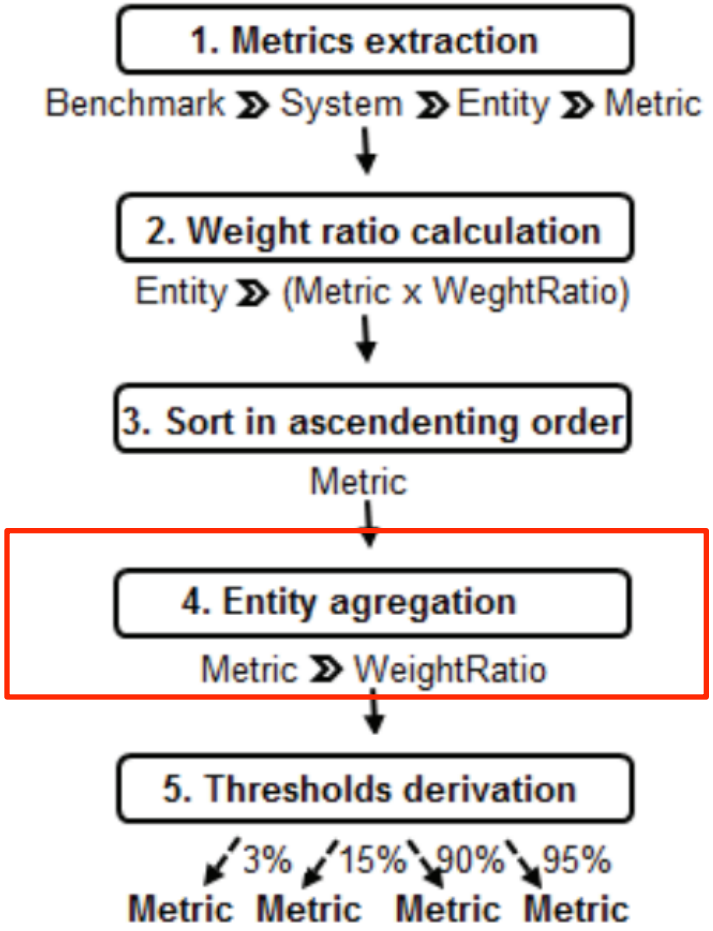
Benchmark
Represents a set of systems

System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

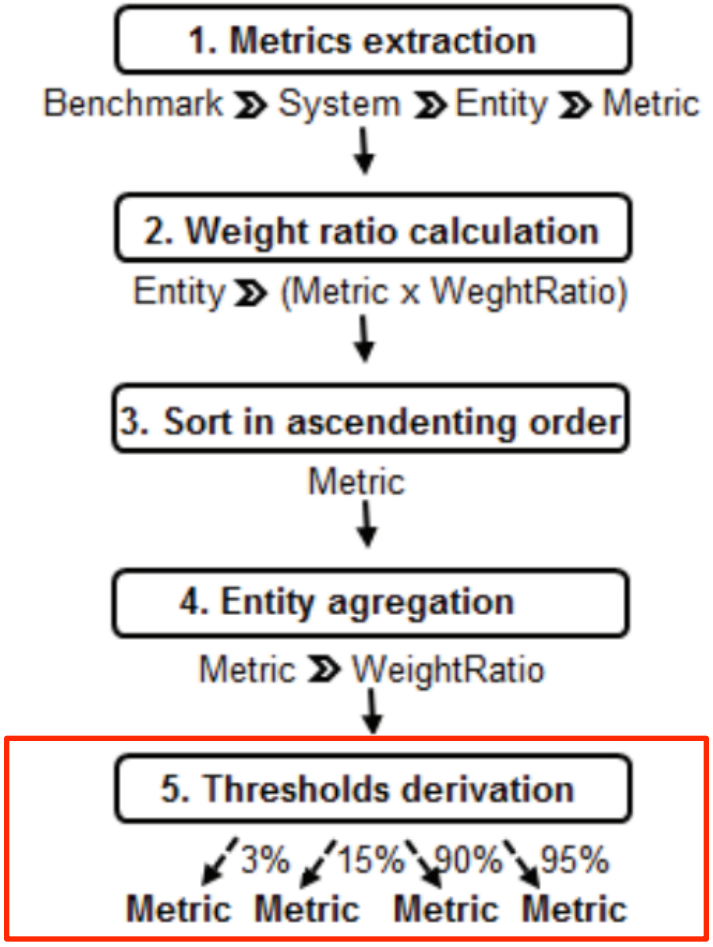
Benchmark
Represents a set of systems

System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Proposed Method



Legend:

»
map relation (one to many relationship)

x
product (pair of columns or elements)

Benchmark
Represents a set of systems

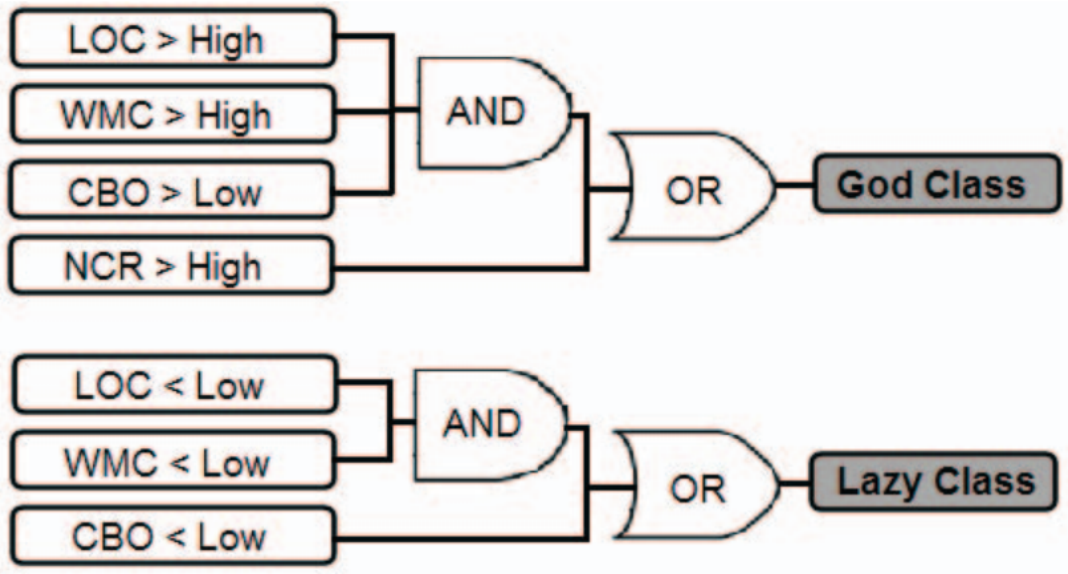
System
Represents individual systems (e.g. MobileMedia)

Entity
Rerepsents a measureble entity (e.g. class)

WeightRatio
Represent the weight percentage inside of the total number of entities (e.g. entity WMC divided by the total WMC of all entites)

Example of Use

- Metric based detection strategy



Example of Use

- 47 SPLs from repositories, filtered 33

	Id	SPL	Tech.	LOC
Benchmark 1	23	IntegerSet [34]	FH-JML	225
	24	UnionFind [34]	FH-JML	194
	25	NumberContractOverrinding [34]	FH-JML	165
	26	NumberConsecutiveContractRef [34]	FH-JML	148
	27	NumberExplicitContractRef [34]	FH-JML	143
	28	BankAccount [34]	FH-JML	122
	29	EPL [12]	AHEAD	98
	30	IntList [34]	FH-JML	94
	31	StringMatcher [34]	FH-JML	45
	32	Stack [34]	FH-Java	22
	33	HelloWorld [12]	AHEAD	22

Example of Use

- 47 SPLs from repositories, filtered 33

	Id	SPL	Tech.	LOC
Benchmarks 1 and 2	15	Elevator [34]	FH-Java	728
	16	ExamDB [34]	FH-JML	568
	17	PokerSPL [34]	FH-JML	461
	18	EmailSystem [34]	FH-Java	460
	19	GPLscratch [34]	FH-JML	405
	20	Digraph [34]	FH-JML	374
	21	MinePump [34]	FH-JML	367
	22	Paycard [34]	FH-JML	319

Example of Use

	Id	SPL	Tech.	LOC
Benchmark 1, 2, and 3	1	BerkeleyDB [34]	FH-Java	37247
	2	AHEAD-Java [1]	AHEAD	16719
	3	AHEAD-guidsl [1]	AHEAD	8738
	4	TankWar [12], [34]	AHEAD	4670
	5	AHEAD-Bali [1]	AHEAD	3988
	6	Devolution [12]	AHEAD	3913
	7	MobileMedia v.7 [14]	AHEAD	2691
	8	WebStore v.6 [14]	AHEAD	2082
	9	DesktopSearcher [12], [34]	AHEAD	1858
	10	GPL [12]	AHEAD	1824
	11	Notepad v.2 [34]	FH-Java	1667
	12	Vistex [34]	FH-Java	1480
	13	GameOfLife [34]	FH-Java	1047
	14	Prop4J [34]	FH-Java	1047

Example of Use

TABLE II. THRESHOLDS VALUES FROM THE PROPOSED METHOD

Benchmark	%	LOC	CBO	WMC	NCR
1	3	3	2	1	1
	15	5	2	2	1
	90	78	12	18	4
	95	139	17	32	8
2	3	3	2	1	1
	15	5	2	2	1
	90	79	12	18	4
	95	143	17	33	9
3	3	3	2	1	1
	15	5	2	2	1
	90	80	13	19	4
	95	147	18	35	9

Evaluation

TABLE VI. RECALL AND PRECISION BASED ON THRESHOLDS DERIVED FROM EACH METHOD

Code Smell	Benchmarks	The Proposed Method		Lanza's Method	
		P	R	P	R
God Class	1, 2, and 3	75.00	42.86	75.00	42.86
Lazy Class	1, 2, and 3	47.62	100	100	10.00

Threats To Validity

- The main Threats
  - Code Smell
    - There many types of smells (Fowler -> 20 types)
    - Not necessarily be a representative
  - Oracle Generation
    - Used to calculate recall and precision
    - Omitted some instances of code smell

Conclusion

- Thresholds that can be used in different contexts.
  - Ex.: identify code smells.
  - Ex.: SIG Model

- Recall for Lazy Class: 100%

Conclusion

- Fits in Requirements:
  - Based on data analysis
  - Strong dependence with the number of Entities
  - Weak dependence with the number of Systems
  - Calculates upper and lower thresholds
  - Thresholds obtained step by step
  - Respect the statistical properties of metrics
  - Systematic, Repeatable, transparent

UFMG

lab-soft
software engineering laboratory

A Method to Derive Metric Thresholds for Software Product Lines

Authors: Gustavo Vale, Eduardo Figueiredo
Presenter: Allan Victor Mori

Software Engineering Lab (LabSoft)
http://labsoft.dcc.ufmg.br/