Table1: LOC Design Metric - Definition and Computation

Lines of Codes (LOC)	
Definition	The number of lines of code of an operation or of a class, including blank lines and comments.
Worse	For greater values.
Computation Details	For **method**: we count the LOC from the method signature to the last curly bracket. For **class**: we count the LOC from the class declaration to the last curly bracket. For **package**: we sum the LOC of the classes declared in the package. For **project**: we sum the LOC of all packages.
Visitor Type	Hybrid Visitor

Implementation details for each entity the visitor can visit

visit: Method	**Visit Type**: AST Visitor
	Applicability: Method, Member of an Annotation
visit: Class	**Visit Type**: AST Visitor
	Applicability: Class, Nested Class, Anonymous Class, Interface, Enumeration, Annotation
visit: Package	**Visit Type**: Model Visitor
	Applicability: Package

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
LOC	Type	Type

visit: Project	**Visit Type**: Model Visitor	
	Applicability: Project	

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
LOC	Package	Package

Table 2: LOCNAMM Design Metric - Definition and Computation

Lines of Codes Without Accessor or Mutator Methods (LOCNAMM)	
Definition	The number of lines of code of a class, including blank lines and comments and excluding accessor and mutator methods and corresponding comments.
Worse	For greater values.
Computation Details	We count the LOC from the class declaration to the last curly bracket.
Visitor Type	AST Visitor

Implementation details for each entity the visitor can visit

visit: Class	**Visit Type**: AST Visitor
	Applicability: Class, Nested Class, Anonymous Class, Interface, Enumeration, Annotation

Table 3: NOPK Design Metric - Definition and Computation

Number of Packages (NOPK)	
Definition	Number of Packages (NOPK)
Worse	-

Computation Details	We declare a dependency to Declared Classes on the en tire project. In this way, we save on the DFMC4J Model all the packages that contain the classes.
Visitor Type	Model Visitor

Implementation details for each entity the visitor can visit

visit: Project	**Visit Type**: Model Visitor
	Applicability: Project

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
Declared Classes	CompilationUnit	Project

Table 4: NOCS Design Metric - Definition and Computation

Number of Classes (NOCS)	
Definition	Total number of classes in a system, in a package or in a class.
Worse	-
Computation Details	For **class**: we sum up the number of nested classes. For **package**: we sum up the NOCS for all the classes in the package. For **project**: we sum up the NOCS for all the packages in the project.
Visitor Type	Model Visitor

Implementation details for each entity the visitor can visit

visit: Class	**Visit Type**: Model Visitor
	Applicability: ComplexType

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
Declared Classes	CompilationUnit	Project
visit: Package	**Visit Type:** Model Visitor	
	Applicability: Package	

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOCS	Type	Package
visit: Project	**Visit Type:** Model Visitor	
	Applicability: Package	

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOCS	Package	Project

Table 5: NOM Design Metric - Definition and Computation

Number of Methods (NOM)	
Definition	NOM represents the number of methods defined locally in a class, counting public as well as private methods. Overridden methods are not taken into account.
Worse	-
Computation Details	For **class**: we sum up the number of Methods Declared In Class. For **package**: we sum up the NOM for all the classes in the package. For **project**: we sum up the NOM for all the packages in the project.
Visitor Type	Model Visitor

Implementation details for each entity the visitor can visit

visit: Class	**Visit Type**: Model Visitor
	Applicability: ComplexType

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
Methods Declared In Class	Type	Type

visit: Package	**Visit Type:** Model Visitor
	Applicability: Package

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOM	Type	Package

visit: Project	**Visit Type:** Model Visitor
	Applicability: Project

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOM	Package	Project

Table 6: NOMNAMM Design Metric - Definition and Computation

Number of Not Accessor or Mutator Methods (NOMNAMM)	
Definition	NOMNAMM represents the number of methods de fined locally in a class, counting public as well as private methods, excluding accessor or mutator methods.
Worse	-
Computation Details	For **class**: we sum up the number of not accessor or mutator Methods Declared In Class. For **package**: we sum up the NOMNAMM for all the classes in the package. For **project**: we sum up the NOMNAMM for all the packages in the project.
Visitor Type	Model Visitor

Implementation details for each entity the visitor can visit

visit: Class	**Visit Type**: Model Visitor
	Applicability: ComplexType

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
Methods Declared In Class	ComplexType	ComplexType

visit: Package	**Visit Type:** Model Visitor
	Applicability: Package

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOMNAMM	Type	Package

visit: Project	**Visit Type:** Model Visitor
	Applicability: Project

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
NOMNAMM	Package	Project

Table 7: NOA Design Metric - Definition and Computation

Number of Attributes (NOA)	
Definition	Number of attributes of a class.
Worse	-
Visitor Type	Model Visitor

Implementation details for each entity the visitor can visit

visit: Class	**Visit Type**: Model Visitor
	Applicability: ComplexType

Dependencies Information:

Dep-visitor	Dep-entity	Dep-level
Attributes of Class	Type	Type