

Format of FindSmells input files

FindSmells uses an input pattern based on the XML markup language. Figure 1 shows the template to create the input data for FindSmells, where each tag used to represent the input data for FindSmells are Metrics, Metric, Values and Value.

```
▼<Metrics scope="name_of_project" type="project">
  ▼<Metric id="name_of_metric" description="full_name_of_metric">
    ▼<Values per="granularity_of_metric">
      <Value name="name_of_entity" source="source_of_entity" package="package_of_entity" value="value_of_metric"/>
    </Values>
  </Metric>
  ▼<Metric id="name_of_metric" description="full_name_of_metric">
    <Value value="value_of_metric"/>
  </Metric>
  . . .
</Metrics>
```

Figure 1. Template for creating the input XML for FindSmells.

- **Metrics:** appears in the first line of the document. This tag informs FindSmells that the information within its scope indicates metrics related to a particular software. Metrics use two attributes to do that: scope and type.
 - ✓ **scope:** refers to the name of the project, whose metrics are referenced.
 - ✓ **type:** attribute default to all input files. It indicates that the metrics in this file refer to a project. For this reason, must contain the word “project”.
- **Metric:** indicates that the values of a certain metric (NOF, NSC, among others) will be informed to the entities of a software. Metric has the following attributes:
 - ✓ **id:** refers to acronym of the metrics, for example: dit, lcom, nsc, etc.
 - ✓ **description:** gives the complete description of the acronym indicated in the attribute “id”, for example: Depth of Inheritance Tree, Lack of Cohesion of Methods, Number of Children, etc.
- **Values:** indicates that there is a list of metric values to the artifacts classes, packages or methods of a software. Values possesses an attribute called per.
 - ✓ **per:** indicates the granularity of metrics in question and whether the artifacts presented are related to: classes, packages or methods. The possible values to this attribute are: “type”, “method” and “packageFragment”.
- **Value:** is used to indicate the value of a metric for an artifact. For class, package and method metrics, this tag must be used within the “Values” tag. For projects metric, the “Values” tag isn’t used, and for this reason, the “Value” tag must be used within tag “Metric”. Values has four attributes: name, source, package and value.

- ✓ **name:** indicates the name of class, package or method measured.
- ✓ **source:** indicates the name of the file, whose artifact is owned by. In the case of classes and methods, this attribute must be filled with the name of the “.java” file indicating where the artifact is located. In the case of packages, this attribute isn’t used.
- ✓ **package:** indicates the full name of package, whose artifact belongs (for example: “documentation.findsmells.xml.input”). For all artifacts, this attribute must be filled out.
- ✓ **value:** indicates the metric value for the respective artifact.

FindSmells supports measures of 23 different metrics. Table 1 shows these metrics.

Metric	Granularity	Description
CA	Package	Afferent Coupling.
CE	Package	Efferent Coupling.
DIT	Class	Depth Inheritance Tree.
LCOM	Class	Lack of Cohesion of Methods.
MLOC	Method	Methods Line of Code.
NBD	Method	Nested Block Depth.
NOC	Package	Number of Classes.
NOF	Class	Number of Attributes.
NOI	Package	Number of Interfaces.
NOM	Class	Number of Methods.
NOP	Project	Number of Packages.
NORM	Class	Number of Overriden Methods.
NSC	Class	Number of Children.
NSF	Class	Number of Static Attributes.
NSM	Class	Number of Static Methods.
PAR	Method	Number of Parameters.
RMA	Package	Abstractness.
RMD	Package	Normalized Distance.
RMI	Package	Instability.
SIX	Class	Specialization Index.
TLOC	Project	Total Lines of Code.
VG	Method	McCabe Cyclomatic Complexity.
WMC	Class	Weighted methods per Class.

Table 1. Metrics supported by the FindSmells tool.