**Assessment Parser Output Format  
03/19/2016 Version 1.3**

This document describes the output format for the parser which transforms the results from Assessment tool specific format to a common format across all the tools, **S**WAMP **C**ommon **A**ssessment **R**esult **F**ormat (SCARF)  
  
**Structure and Format of the Parser Output**

The parser output is a XML file, for which the root tag is *AnalyzerReport*

and has following property attributes.

|  |  |  |
| --- | --- | --- |
| **<AnalyzerReport> Property Attributes** | **Description** | **Mandatory** |
| tool\_name | Name of the Software Assessment tool | Yes |
| tool\_version | Version number of the Software Assessment tool | Yes |
| uuid | Unique Id extracted from build\_summary.xml output from a SwA run | Yes |

**E.g.**

*<AnalyzerReport tool\_name="Find Bugs" tool\_version="2.0.2" uuid="125edfbf-cba8-4a68-af22-3687879ed232" build\_id="5">*

***<AnalyzerReport>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<AnalyzerReport> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <BugInstance> | See *<BugInstance>* section. | No | Yes |
| <BugSummary> | See *<BugSummary>* section. | No | No |
| <Metric> | See *<Metric>* section. | No | Yes |
| <MetricSummaries> | See *<MetricSummaries>* section. | No | No |

The root tag **AnalyzerReport** has ***BugInstance*** as child element, which contains a property named “**id”**. There can be multiple occurrence of *BugInstance*, one per each instance of bug from the SwA tools. The “id” attribute in the BugInstance tag is a unique integer id generated by the Assessment Parser tool.

**E.g.**  
<*BugInstance id*="1">

***<BugInstance>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<BugInstance> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <ClassName> | Fully qualified name of the class where bug was found. | No | No |
| <Methods> | See *<Methods>* section. | No | No |
| <BugLocations> | See *<BugLocations>* section. | Yes | No |
| <BugGroup> | Tool specific Bug Grouping/Category information. | No | No |
| <BugRank> | Tool specific bug ranking information (provided by the tool). | No | No |
| <BugSeverity> | Severity of the reported bug. | No | No |
| <CweId> | Tool Specific CWE Id for the reported bug. | No | Yes |
| <BugMessage> | Message from the tool explaining the reported bug. | Yes | No |
| <ResolutionSuggestion> | Any suggestion/hint from the tool for resolving the bug. | No | No |
| <BugTrace> | See *<BugTrace>* section. | Yes | No |
| <BugCode> | Contains abbreviation describing the specific bug | Yes | No |

**<Methods> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<Methods> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Method> | Name of the method where bug is reported. See *<Method>* Attributes section for further detail. | No | Yes |

**<Method> Element Attributes**

|  |  |  |
| --- | --- | --- |
| **<Method> Element Attributes** | **Description** | **Mandatory** |
| id | Unique integer id generated by the parser per method. | Yes |
| primary | contains ‘*true*’ value if this is the primary function where bug is reported, else the value is ‘*false*’ | Yes |

**<BugLocations> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<BugLocations> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Location> | Contains details describing the location of the Bug. See <Location> section for more details. | Yes | Yes |

**<Location> Element Attributes**

|  |  |  |
| --- | --- | --- |
| **<Location> Element Attributes** | **Description** | **Mandatory** |
| id | Unique integer id generated by the parser per bug location. | Yes |
| primary | contains ‘*true*’ value if this is the primary location of the bug, else the value is ‘*false*’ | Yes |

**<Location> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<Location> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <SourceFile> | Path of the source code file starting from the package name. | Yes | No |
| <StartLine> | Start line number for the bug location in the source file | No | No |
| <EndLine> | Start line number for the bug location in the source file, if the location for the bug is a single line, then end line is same as start line. | No | No |
| <StartColumn> | Start column number within a line for the bug. | No | No |
| <EndColumn> | End column number within a line for the bug. | No | No |
| <Explanation> | Message from the tool specific to the location. | No | No |

**<BugTrace> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<BugTrace> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <BuildId> | This is the build artifacts id extracted from build\_summary.xml file | Yes | No |
| <AssessmentReportFile> | Path to the original file generated by the assessment tool. | Yes | No |
| <InstanceLocation> | Location of the bug instance within the original file. See <InstanceLocation> section | No | No |

**<InstanceLocation> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<InstanceLocation> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Xpath> | For where raw result file is in XML format , it provides the Xpath to the instance in the file. | No | No |
| <LineNum> | For where the raw result file is simple text file, it provides the line number for the bug instance. See <LineNum> section. | No | No |

**<LineNum> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<LineNum> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Start> | Provides the start line number for the bug instance in the file | Yes | No |
| <End> | Provides the end line number for the bug instance in the file | Yes | No |

The root tag **AnalyzerReport** has ***BugSummary*** as child element. There can be only one occurrence of *BugSummary*, if and only if there is at least one *BugInstance.*

**<BugSummary> Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<BugSummary> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <BugCategory> | Provides summary of one category of bug. See <BugCategory> section. | Yes | Yes |

**<BugCategory> Element Attributes**

|  |  |  |
| --- | --- | --- |
| **<BugCategory> Element Attributes** | **Description** | **Mandatory** |
| group | Provides tool specific group of this bug type. | Yes |
| code | Provides specific bug code for this bug type. | Yes |
| count | Provides total number of bugs found with this type | Yes |
| bytes | Total bytes this bug type occupies in the SCARF file. | Yes |

The root tag **AnalyzerReport** has ***Metric*** as child element, which contains a property named “**id”**. There can be multiple occurrence of *Metric*, one per each instance of metric from the SwA tools. The “id” attribute in the Metric tag is a unique integer id generated by the Assessment Parser tool.

***<Metric>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<Metric> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Location> | Contains location of the source file. See <Location> section for more details. | Yes | No |
| <Class> | Name of class being analyzed | No | No |
| <Method> | Name of the method being analyzed. | No | No |
| <Type> | Indicates the type of metric being used. | Yes | No |
| <Value> | Indicates the results of the given metric. | Yes | No |

***<Location>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<Location> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <SourceFile> | Contains location of the source file. | Yes | No |

The root tag **AnalyzerReport** has ***MetricSummaries*** as child element. There can be only one occurrence of *MetricSummaries*, if and only if there is at least one Metric.

***<MetricSummaries>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<MetricSummaries> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <MetricSummary> | Contains a summary of one metric types results. See <MetricSummary> section for more details. | Yes | Yes |

***<MetricSummary>* Child Elements**

|  |  |  |  |
| --- | --- | --- | --- |
| **<MetricSummary> Child Elements** | **Description** | **Mandatory** | **More than one** |
| <Type> | Indicates the type of metric in this summary. | Yes | No |
| <Count> | Indicates the total number of metrics of this type. | Yes | No |
| <Sum> | Contains the sum of the values from all metrics of this type. | Yes | No |
| <SumOfSquares> | Provides the sum of squares value from all of the values of this type | Yes | No |
| <Minimum> | Provides the lowest value received on a metric of this type. | Yes | No |
| <Maximum> | Provides the highest value received on a metric of this type. | Yes | No |
| <Average> | Provides an average value of all the metrics of this type. | Yes | No |
| <StandardDeviation> | Provides the standard deviation on the values of the metrics of this type. | Yes | No |

**Availability of extractable information from assessment tools**

Assessment tool covered for the above output XML format are:

|  |  |
| --- | --- |
| **Assessment Tool** | **Language** |
| Find Bugs | Java |
| PMD | Java |
| Clang Static Analyzer | C/C++ |
| Cpp Check | C/C++ |
| Gcc Warnings | C/C++ |

Information Mapping:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Element** | **Find Bugs** | **PMD** | **Clang Static Analyzer** | **Cpp Check** | **GCC** |
| Classname | ✔ | ✔ |  |  |  |
| Methods/Method | ✔ | ✔ |  |  |  |
| BugLocations/Location/SourceFile | ✔ | ✔ | ✔ | ✔ | ✔ |
| BugLocations/Location/StartLine | ✔ | ✔ | ✔ | ✔ | ✔ |
| BugLocations/Location/EndLine | ✔ | ✔ | ✔ | ✔ | ✔ |
| BugLocations/Location/StartColumn |  | ✔ |  |  | ✔ |
| BugLocations/Location/EndColumn |  | ✔ |  |  | ✔ |
| BugLocations/Location/Explanation | ✔ |  | ✔ | ✔ |  |
| BugGroup | ✔ | ✔ | ✔ | ✔ | ✔ |
| BugSeverity | ✔ | ✔ |  | ✔ |  |
| BugRank | ✔ |  |  |  |  |
| CweId | ✔(for some bugs) |  |  |  |  |
| BugMessage | ✔ | ✔ | ✔ | ✔ | ✔ |
| ResolutionSuggestion | ✔ |  |  |  |  |
| BugTrace/AssessmentReportFile | ✔ | ✔ | ✔ | ✔ | ✔ |
| BugTrace/InstanceLocation/XPath | ✔ | ✔ |  | ✔ |  |
| BugTrace/InstanceLocation/LineNum/Start |  |  |  |  | ✔ |
| BugTrace/InstanceLocation/LineNum/End |  |  |  |  | ✔ |

**Sample Parsed XML - SWAMP Common Assessment Result Format (SCARF)**

<AnalyzerReport tool\_name="findbugs" tool\_version="2.0.2" uuid="fc4a922e-faa9-4a2d-b12b-4f5020f96c5c">

<BugInstance id="1">

<ClassName>org.apache.lucene.codecs.BlockTreeTermsReader$FieldReader$IntersectEnum</ClassName>

<Methods>

<Method id="1" primary="true">&lt;init></Method>

</Methods>

<BugLocations>

<Location id="1" primary="true">

<SourceFile>solr-4.5.0/lucene/core/src/java/org/apache/lucene/codecs/BlockTreeTermsReader.java</SourceFile>

<StartLine>851</StartLine>

<EndLine>851</EndLine>

<Explanation>Dereferenced at BlockTreeTermsReader.java:[line 851]</Explanation>

</Location>

<Location id="2" primary="false">

<SourceFile>solr-4.5.0/lucene/core/src/java/org/apache/lucene/codecs/BlockTreeTermsReader.java</SourceFile>

<StartLine>838</StartLine>

<EndLine>838</EndLine>

<Explanation>Null value at BlockTreeTermsReader.java:[line 838]</Explanation>

</Location>

<Location id="3" primary="false">

<SourceFile>solr-4.5.0/lucene/core/src/java/org/apache/lucene/codecs/BlockTreeTermsReader.java</SourceFile>

<StartLine>839</StartLine>

<EndLine>839</EndLine>

<Explanation>Known null at BlockTreeTermsReader.java:[line 839]</Explanation>

</Location>

</BugLocations>

<BugGroup>CORRECTNESS</BugGroup>

<BugCode>NP\_NULL\_ON\_SOME\_PATH</BugCode>

<BugRank>6</BugRank>

<BugSeverity>1</BugSeverity>

<BugMessage>Possible null pointer dereference of BlockTreeTermsReader$FieldReader.index in new org.apache.lucene.codecs.BlockTreeTermsReader$FieldReader$IntersectEnum(BlockTreeTermsReader$FieldReader, CompiledAutomaton, BytesRef)</BugMessage>

<ResolutionSuggestion> &lt;p> There is a branch of statement that, &lt;em>if executed,&lt;/em> guarantees that a null value will be dereferenced, which would generate a &lt;code>NullPointerException&lt;/code> when the code is executed. Of course, the problem might be that the branch or statement is infeasible and that the null pointer exception can't ever be executed; deciding that is beyond the ability of FindBugs. &lt;/p> </ResolutionSuggestion>

<BugTrace>

<BuildId>1</BuildId>

<AssessmentReportFile>/home/bhaskar/results/assessment\_report1-1.xml</AssessmentReportFile>

<InstanceLocation>

<Xpath>/BugCollection/BugInstance[5]</Xpath>

</InstanceLocation>

</BugTrace>

</BugInstance>

</AnalyzerReport>