**Martin packaging metrics**

The Martin packaging metrics or software packaging metrics were mentioned by Robert Cecil Martin in “Agile software development: principles, patterns, and practices” and can be subdivided into 5 different metrics: Afferent couplings (Ca), Efferent couplings (Ce), Abstractness (A), Instability (I), and Distance (D).

**Afferent couplings**

The afferent couplings metric measures the number of classes in other packages that depend upon classes within the package, this metric is an indicator of the package’s responsibility (Afferent couplings signal inward).

|  |  |
| --- | --- |
| Ca | Package |
| 13 210 | org.jabref.model.entry.field |
| 10 388 | org.jabref.model.entry |
| 4 109 | org.jabref.logic.l10n |

These are the packages that are most responsible, this means that a lot of classes from other packages are dependent on classes in this package.

**Efferent couplings**

The afferent couplings metric measures the number of classes in other packages that the classes in a package depend upon, this metric is an indicator of the package’s dependence on externalities (Efferent couplings signal outward).

|  |  |
| --- | --- |
| Ce | Package |
| 3 133 | org.jabref.model.entry.types |
| 2 707 | org.jabref.logic.importer.fileformat |
| 1 901 | org.jabref.gui |

These are the packages that are most dependent on externalities, this means that they depend on a lot of classes that are from other packages.

**Abstractness**

The abstractness metric measures the ratio of the number of abstract classes (and interfaces) in the analyzed package to the total number of classes in the analyzed package. The range for this metric is 0 to 1, with A=0 indicating a completely concrete package and A=1 indicating a completely abstract package.

|  |  |
| --- | --- |
| A | Package |
| 1,00 | org.jabref.architecture |
| 1,00 | org.jabref.logic.preview |
| 1,00 | org.jabref.model.openoffice.backend |
| … | … |
| 0,00 | org.jabref.gui.bibtexextractor |
| 0,00 | org.jabref.gui.auximport |
| 0,00 | org.jabref.cli |

These are the packages with the most and least Abstractness. The packages with A=1 are 100% abstract (only abstract classes and interfaces) and the packages with A=0 are 100% concrete (no abstract classes or interfaces).

**Instability**

The Instability metric measures the ratio of efferent coupling to total coupling (I = Ce / (Ce + Ca)), this is an indicator of the package's resilience to change. This metric ranges from 0 to 1, with I=0 indicating a completely stable package and I=1 indicating a completely unstable package.

|  |  |
| --- | --- |
| I | Package |
| 1,00 | org.jabref.architecture |
| 1,00 | org.jabref.gui.logging |
| 1,00 | org.jabref.logic.openoffice.action |
| … | … |
| 0,00 | org.jabref.gui.util.uithreadaware |
| 0,00 | org.jabref.logic.help |
| 0,00 | org.jabref.gui.util.uithreadaware |

These are the packages that are most and least resilient to change, this means that in the most instable classes most of the couplings are Efferent and in the least instable classes most of the couplings are Afferent.

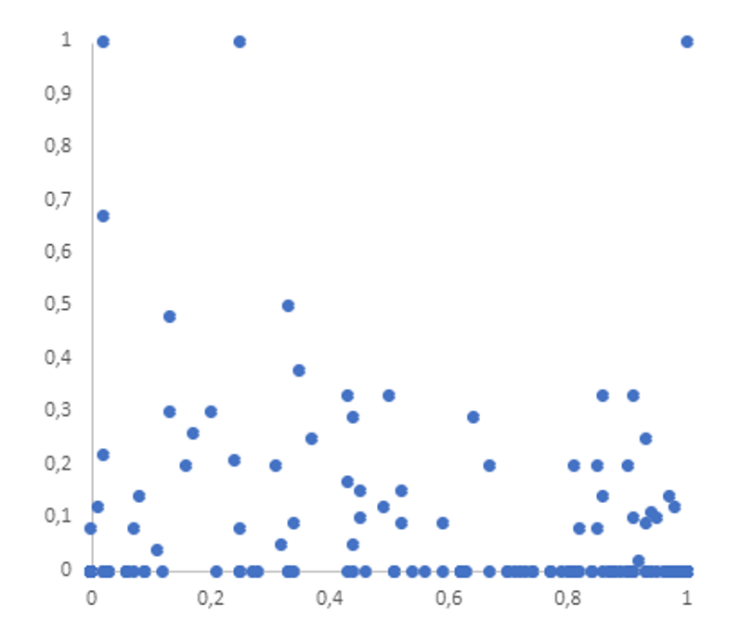
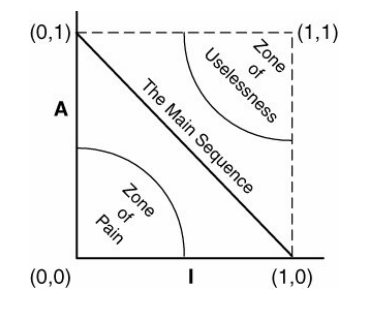
**Distance**

The distance metric measures the perpendicular distance of a package from the idealized line A + I = 1. D is calculated as D = |A + I - 1|, this metric is an indicator of the package's balance between abstractness and stability. Ideal packages are either completely abstract and stable (I=0, A=1) or completely concrete and unstable (I=1, A=0). The range for this metric is 0 to 1, with D=0 indicating a package that is coincident with the main sequence and D=1 indicating a package that is as far from the main sequence as possible.

|  |  |  |  |
| --- | --- | --- | --- |
| D | A | I | Package |
| 1,00 | 0,00 | 0,00 | org.jabref.gui.util.uithreadaware |
| 1,00 | 0,00 | 0,00 | org.jabref.logic.help |
| 1,00 | 0,00 | 0,00 | org.jabref.logic.journals |

These are the most distant packages from the main sequence, this means that they either have mainly abstract classes and Efferent couplings (dependent on externalities) or mainly non abstract classes and Afferent couplings (other packages are dependent on this one).

Packages with A=0 and I=0 are highly stable and concrete, such a component is not desirable because it is rigid. It cannot be extended because it is not abstract and it is very difficult to change because of its stability, this can cause a Shotgun Surgery code smell.

Packages with A=1 and I=1 are undesirable because they are maximally abstract and yet have no dependents, such components are useless and can cause a Lazy Class or Speculative Generality code smell.

A

Distance of the components of the project

I