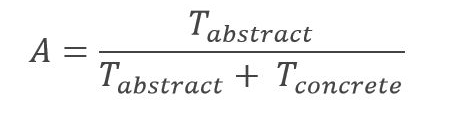
Martin Packaging Metrics Report

A métrica Martin Packaging tem como objetivo identificar packages mal organizadas/planificadas e assim melhorar o desenvolvimento de software.

A métrica Martin Packaging tem 5 parâmetros de avaliação:

**A – Abstractness**

Esta métrica é utilizada para medir o grau de abstração dos packages. Indo pela definição, a abstração é o número de classes abstratas no package para o número de todas as classes.



Onde:

Tabstract – Número de classes abstratas num package;

Tconcrete – Número de classes concretas num package.

Os valores considerados normais para esta métrica deveriam ser valores entre 0 e 1. Packages que são considerados estáveis (outra métrica que irá ser abordada mais tarde no relatório), ou seja, que têm uma dependência pequena noutros packages, também deveriam ser abstratas (ter um valor de A próximo de 1). Noutro caso, os packages mais instáveis deveriam consistir em classes concretas (ter um valor de A próximo de 0).

**Ca – Afferent Coupling**

Esta métrica é utilizada para medir tipos de dependências entre os packages. Permite-nos medir a sensibilidade dos outros packages, às mudanças do package observado.

Os valores considerados normais desta métrica são valores entre o 0 e o 500.

Valores altos nesta métrica sugerem uma alta estabilidade do componente estudado. Isto porque a classe depende de muitas outras classes e, por isso, não pode ser modificada significativamente, pois a probabilidade de espalhar as mudanças é muito grande.

**Ce – Efferent Coupling**

Esta métrica é utilizada para medir relações entre classes. É o número de classes num package dado, que dependem de outras classes num outro package. Permite-nos medir a vulnerabilidade dos packages, em relações a mudanças, em packages que esse depende.

Os valores considerados normais nesta métrica são valores entre 0 e 20, valores mais altos que isto criam problemas de desenvolvimento de código.

Um valor alto neste tipo de métrica (**Ce** > 20) indica instabilidade do package, qualquer tipo de mudança de classes externas pode causar a necessidade de fazer alterações neste package.

**D – Normalized Distance from Main Sequence**

Esta métrica é utilizada para medir o balanço entre a estabilidade (métrica que irá ser abordada mais tarde no relatório) e abstração. É calculada da seguinte maneira:



Onde:

A – Abstração;

I – Instabilidade.

O valor desta métrica devia ser o valor mais baixo possível, para que os componentes estivessem perto da sequência principal.

**I – Instability**

Esta métrica é utilizada para medir a suscetibilidade das classes mudarem. De acordo com a definição, instabilidade é a proporção das **Ce** para todas as dependências de packages.

A instabilidade é calculada através de:



Onde:

**Ce** - **Efferent Coupling**;

**Ca** - **Afferent Coupling**.

Os valores considerados normais para esta métrica deveriam ser valores entre 0 e 1.

E dentro desses valores deveríamos ter packages instáveis ou estáveis, ou seja, devíamos evitar packages com uma estabilidade intermédia.

**Verificação dos valores obtidos ao calcular métrica no trabalho**

Começando por verificar a **Abstractness** (**A**):

- Através dos dados recolhidos, conseguimos verificar que a maior parte dos packages têm um valor baixo de abstração, significando que a maior parte deles têm um nível baixo de dependência de outros packages. Alguns dos packages preocupantes serão aqueles que se encontram com valor aproximado do meio (apróx. 0.5), pois dessa forma estes packages terão mudanças que poderão ser difíceis.

- Alguns dos packages problemáticos serão packages como:

Uma imagem com texto

Descrição gerada automaticamente

Passando agora para **Afferent Coupling** (**Ca**):

- Ao verificar os valores, verifica-se que a maioria dos packages têm um valor baixo de **Ca**, o que é positivo, pois significa que a maior parte dos packages não tem dependência de outros, pelo que ao mudá-los trará uma alteração significativa dentro do código do package.

- Uns dos packages mais problemáticos serão o **net.sourceforce.ganttproject.task** e **net.sourceforge.ganttproject**, o que ao analisar os packages faz sentido, sendo estes os packages que se vai desenvolver mais software e que se vai tratar da estrutura de dados da aplicação, ao desenvolver código nestes packages, que são dependentes de todos os outros, simplesmente irão se notar as diferenças nesses mesmos packages, uma vez que os outros já foram contruídos/implementados com intenção de não alterar muito ao longo do desenvolvimento de novo software na aplicação.

**Efferent Coupling** (**Ce**):

- Verificando os valores recolhidos observamos que a média de valores encontra se muito elevada comparado com o que é proposto (**Ce** <= 20), pelo que isto significa que a aplicação poderá ter muitos problemas de desenvolvimento de software no futuro. Este tipo de valores indica que o projeto encontra-se com valor elevado de instabilidade de packages e qualquer tipo de mudança de classes, externas aos packages referentes, poderá implicar um elevado número de alterações a esses mesmos packages externos.

- Alguns dos packages problemáticos serão packages como:



Estes packages encontram-se com os valores de **Ce** mais elevados, pelo que com qualquer tipo de alteração externa, teria uma chance muito elevada de ser preciso alterar código nestes packages, mas ao analisar os mesmo verificamos que estes packages fazem parte do pequeno número que vai ser preciso alterar código, pelo que, neste projeto, os problemas de acordo com esta métrica não vão ser muito notados.

**Normalized Distance from Main Sequence** (**D**):

* Com os valores obtidos observamos que metade dos packages do projeto têm um valor extremamente baixo, ou seja, têm um bom balanço entre instabilidade e abstração.

- Alguns dos packages problemáticos serão packages como:

Uma imagem com texto

Descrição gerada automaticamente

Ao analisar estes packages mais profundamente e com auxílio de **I(Instability)** e **A(Abstractness)** verificamos que temos uma situação que não é muito agradável, ambos os packages têm o valor de **I** e de **A** de 0, pelo que significa que os packages são extremamente estáveis e concretos, ou seja, nada flexíveis e não podem ser estendidos.

**Instability** (**I**):

- Através dos dados recolhidos, conseguimos verificar que a maior parte dos packages têm um valor elevado de Instabilidade, o que faz sentido, comparado com os resultados obtidos em **A(Abstractness)**, significando assim que os packages são considerados instáveis, pois serão facilmente feitas mudanças no código que implicam alterar esses packages. Também verificamos que das restantes, a maior parte tem um valor baixo de Instabilidade, pelo que significa que esses packages serão mais difíceis de serem alterados devido a sua importância (estabilidade).

- Alguns dos packages problemáticos serão packages como:

Uma imagem com texto

Descrição gerada automaticamente

Estes packages encontram-se com valores próximos de 0.5. Estes packages são considerados packages de estabilidade intermédia, pelo não fornecem uma boa solução para desenvolvimento de software.

Gonçalo Cerveira

Nrº 61696