**Dependency Metrics** Nádia Mendes 53175

**Summary**:

- CYCLIC (Number of cyclic dependencies) measures, for each class c, the number of classes c directly depends on, and that in turn depend on c.

-DCY (Number of dependencies) measures, for each class c, the number of classes c directly depends on.

-DCY\* (Number of transitive dependencies) measures, for each class c, the number of classes c directly or indirectly depends on.

-DPT (Number of dependants) measures, for each class c, the number of classes that directly depend on c.

-DPT\* (Number of transitive dependants) measures for each class c, the number of classes that directly or indirectly depend on c.

-PDCY (Parse distance or Parse depth) this variable generally measures the distance or depth in the parse tree between two elements in a sentence. This metric is used to evaluate how far apart or close together elements are in the sentence structure.

-PDPT (Parse tree depth) measures the depth of an element in the parse tree of a sentence.

**Data Visualization –**

**Uma imagem com texto, captura de ecrã, software, Software de multimédia

Descrição gerada automaticamente**

**Top 5 CYCLIC –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.server.networking.Server 805 63 989 61 923 6 12

net.sf.freecol.server.networking.DummyConnection 805 3 989 10 923 2 3

net.sf.freecol.server.model.TimedSession 805 2 989 2 923 2 1

net.sf.freecol.server.model.Session 805 3 989 2 923 2 2

net.sf.freecol.server.model.ServerUnit 805 5 989 3 923 3 3

Uma imagem com texto, captura de ecrã, número, Gráfico

Descrição gerada automaticamente

**Top 5 DCY –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.server.control.InGameController 805 118 989 83 923 11 1

net.sf.freecol.client.gui.Widgets 805 111 989 2 923 10 12

net.sf.freecol.common.networking.ServerAPI 805 104 989 10 923 3 5

net.sf.freecol.server.model.ServerPlayer 805 96 989 124 923 9 1

net.sf.freecol.client.gui.SwingGUI 805 95 989 1 923 20 9

Uma imagem com texto, captura de ecrã, Gráfico, número

Descrição gerada automaticamente

**Top 5 DCY\* –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.AllTests 0 5 1083 0 0 5 0

net.sf.freecol.common.AllTests 0 5 1050 1 1 5 1

net.sf.freecol.common.model.AllTests 0 38 1040 1 2 1 1

net.sf.freecol.server.AllTests 0 4 1022 1 1 4 1

net.sf.freecol.server.ai.AllTests 0 8 1011 1 2 2 1

Uma imagem com texto, captura de ecrã, número, Gráfico

Descrição gerada automaticamente

**Top 5 DPT –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.common.model.FreeColObject 805 5 1083 380 923 4 34

net.sf.freecol.common.model.Player 805 5 1050 335 923 7 26

net.sf.freecol.common.model.Game 805 38 1040 328 923 7 27

net.sf.freecol.common.model.Unit 805 4 1022 297 923 6 25

net.sf.freecol.client.FreeColClient 805 8 1011 294 923 15 16

Uma imagem com texto, captura de ecrã, número, Gráfico

Descrição gerada automaticamente

**Top 5 DPT\* –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.common.util.Utils 0 1 1 67 937 0 20

net.sf.freecol.common.util.StringUtils 0 1 1 53 936 0 17

net.sf.freecol.common.resources.Resource 0 0 0 16 935 0 3

net.sf.freecol.common.util.CachingFunction 0 0 0 3 933 0 2

net.sf.freecol.common.util.CollectionUtils 0 3 3 165 932 1 32

Uma imagem com texto, captura de ecrã, número, Gráfico

Descrição gerada automaticamente

**Top 5 PDCY –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.client.gui.SwingGUI 805 95 989 1 923 20 1

net.sf.freecol.server.FreeColServer 805 59 989 138 923 15 15

net.sf.freecol.common.debug.DebugUtils 805 64 989 8 923 15 5

net.sf.freecol.client.gui.GUI 805 71 989 153 923 15 15

net.sf.freecol.client.gui.FreeColClient 805 37 989 294 923 15 16

Uma imagem com texto, captura de ecrã, Tipo de letra, Gráfico

Descrição gerada automaticamente

**Top 5 PDPT –** CYCLIC DCY DCY\* DPT DPT\* PDCY PDPT

net.sf.freecol.common.model.FreeColObject 805 17 989 338 923 4 34

net.sf.freecol.common.util.CollectionUtils 0 3 3 165 923 1 32

net.sf.freecol.common.model.Specification 805 65 989 262 923 5 29

net.sf.freecol.common.model.Game 805 50 989 328 923 7 27

net.sf.freecol.common.model.Player 805 81 989 325 923 7 26

Uma imagem com texto, captura de ecrã, Gráfico, file

Descrição gerada automaticamente

**Discussion-**

As can be seen in the first figure 2, the 5 classes with the highest Number of cyclic dependencies (CYCLIC) are contained in the net.sf.freecol.server.networking package and the net.sf.freecol.server.model package, these present a high number of cyclic dependencies, indicating that the project is complex, with many dependency relationships between its components. Which makes the code more difficult to understand and modify.

As can be seen in the first figure 2, the 5 classes with the highest Number of cyclic dependencies (CYCLIC) are contained in both the net.sf.freecol.server.networking package and the net.sf.freecol.server.model package, These present a high number of cyclic dependencies, indicating that the project is complex, with many dependency relationships between its components. Which makes the code more difficult to understand and modify.

It can also be seen that due to the existence of so many cyclical dependencies, the project becomes more difficult to test, since it is more difficult to isolate the components for unit testing. Which can impact the quality of the software.

A possible solution would be to refactor the project, which would involve restructuring the code to reduce or eliminate cyclical dependencies, thus making the project easier to modulate and maintain.

Regarding the number of dependencies by class (DCY), we can conclude that there are classes that present very high values, which suggests that there is a high level of coupling between the project components, as is the case with the classes identified above, however it is verified that there is also the existence of classes whose value is relatively low or even null. The average is 10.95, which suggests that it is a relatively good value.

When we analyze the number of transitive dependencies (DCY\*) we see that in this project there is a high number of classes with a high value, and on average each class has around 817.49 indirect dependencies in relation to other components, thus it appears that in these classes the code is more difficult to understand and manage. We also verified that once again due to this relatively high value the project becomes difficult to maintain since with many transitive dependencies changes to an indirect dependency can affect many components, requiring extensive testing and validation, and this high value can also affect project performance, since more resources may be required to load and manage all dependencies.

Regarding the number of dependents (DPT), with an average of 10.48, we can see that in terms of direct dependencies, the project has a moderate level of coupling.

Regarding the number of transitive dependencies (DPT\*), we found an average of 807.06, this value is significantly high, which suggests that, in addition to direct dependencies, there are many transitive dependencies (i.e., indirect dependencies, dependency dependencies). This indicates that changes in one component can potentially affect many other components, including those that indirectly depend on the component in question. The value of Parse distance or Parse depth has an average of 3.61, which indicates that on average the syntactic analysis of the code reaches a depth of 3.61 levels, therefore we can consider it as a moderate depth, which suggests that the code does not present excessive complexity in terms of analysis structure.

Finally, we verified that the Parse tree deth (PDPT) value presented a value of 2.78, which indicates that the parse tree structure is not very deep, which is positive, as excessive depth would make the code more difficult to understand.