**Dependency metrics report**

With the help of some tables generated with the help of a script (script.ts) we can try to make sense of the dependency data generated by the plugin MetricsReloaded.

Let’s start by analyzing the class dependencies data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Class metrics** | | | | | |
| **Metrics** | **Average** | **Maximum** | | **Minimum** | |
| **value** | **count** | **value** | **count** |
| Number of cyclic dependencies | 296.7709924 | 517 | 451 | 0 | 297 |
| Number of dependencies | 6.812977099 | 101 | net.sourceforge.ganttproject.GanttProject | 0 | 99 |
| Number of transitive dependencies | 566.1068702 | 888 | 2 | 0 | 99 |
| Number of dependents | 4.117048346 | 118 | net.sourceforge.ganttproject.language.GanttLanguage | 0 | 40 |
| Number of transitive dependents | 506.9185751 | 659 | biz.ganttproject.core.option.ChangeValueEvent | 0 | 40 |
| Number of package dependencies | 3.069974555 | 30 | net.sourceforge.ganttproject.GanttProject | 0 | 107 |
| Number of dependent packages | 2.213740458 | 32 | net.sourceforge.ganttproject.language.GanttLanguage | 0 | 40 |

As we can see there is a very high number of averages of cyclic dependencies. Taking in account that there were considered 786 classes it means that each class depends on about 40% of the other classes which in turns depends on them (directly or indirectly).

Another thing to take notice is that there are 451 classes (about 57% classes) that depends cyclically on 517 (about 66% of the classes).

Another interesting thing is that the class net.sourceforge.ganttproject.GanttProject.java the class with more dependencies which makes sense since it’s the main class.

We can also notice that in average a class is a decency of other 4 classes (which means that if we draw a dependency graph every node would have 4 edges in average) and that the class net.sourceforge.ganttproject.language.GanttLanguage.java is the one that more classes depend on. This might suggest that probably there are some code smells there, and it is also a good place to start looking if one ever decides to refactor this codebase.

The class with more package dependencies is the main class which totally makes sense.

And the class that more packages depend on is that one that more classes depend on, which also makes sense.

For what we’ve seen so far it seems that the principle of low coupling is not well respected by the classes, they depend too much in each other and there are classes that depends (which means they know) about more than half of the classes that exists in the project.

According to the data as well, in average every class is known by other 4 class and in average every class knows 6 classes. This leads us to conclude that it’s probably very hard to understand the code base from scratch (which may reduce its lifetime), to extend it, if one wish to, and to change anything because if we change in one class we probably will have to touch other 6 class which in turn could make us change another 6 until we’ve changed every class in the project.

Let’s take a look on the interface dependency data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Interface metrics** | | | | | |
| **Metrics** | **Average** | **Maximum** | | **Minimum** | |
| **value** | **count** | **value** | **count** |
| Number of cyclic dependencies | 193.5865922 | 517 | 67 | 0 | 104 |
| Number of dependencies | 1.927374302 | 23 | net.sourceforge.ganttproject.task.TaskManager | 0 | 49 |
| Number of transitive dependencies | 339.396648 | 877 | net.sourceforge.ganttproject.task.CustomColumsListener | 0 | 49 |
| Number of dependents | 13.05027933 | 186 | net.sourceforge.ganttproject.task.Task | 0 | 3 |
| Number of transitive dependents | 560.3519553 | 658 | biz.ganttproject.core.option.ChangeValueListener | 0 | 3 |
| Number of package dependencies | 1.279329609 | 9 | net.sourceforge.ganttproject.task.TaskManager | 0 | 49 |
| Number of dependent packages | 5.206703911 | 35 | net.sourceforge.ganttproject.gui.UIFacade | 0 | 3 |

After seeing the result from the class metrics, it doesn’t surprise us that the interfaces are also a bit accoupled. What we can notice is that the number of dependents class (in this case or interface) has raised to 13. A good thing is that the in average an interface only depend on about 2 class/interfaces.

We can notice that there are some interfaces that are a bit coupled with class/interfaces, for example we can notice that 67 interfaces or about 37% (knowing that there were considered 179 interfaces) depends cyclically on another 517 interfaces/class that is about 53% of the interfaces + the classes considered in this metric. In the middle of this there is a positive point, which is 104 interfaces (about 58%) aren’t in any cyclic dependency, which means that they are fine, and we can just look of the ones with that are in and see what can be done.

We can also notice that 35 packages (about 43% of them) depends on the interface UIFacade which I can’t describe as good or that bad, because I think that it at some point reduced the number of interfaces that those packages would depend on, I can honestly say that this number is still high.

Before we start looking at the package’s dependencies, I just wanted to make a point. So far, we can notice that there is more dependency between class than between interfaces. Also, that there is probably more classes/interface depending on classes than interfaces which in turn isn’t a good thing.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Package metrics** | | | | | |
| **Metrics** | **Average** | **Maximum** | | **Minimum** | |
| **value** | **count** | **value** | **count** |
| Number of cyclic package dependencies | 33.2 | 51 | 52 | 0 | 24 |
| Number of package dependencies | 7.6625 | 54 | net.sourceforge.ganttproject | 0 | 11 |
| Number of dependent packages | 7.6625 | 47 | net.sourceforge.ganttproject | 0 | 7 |
| Number of transitively dependent packages | 52.83544304 | 69 | org.w3c.util | 1 | 6 |

Finally, we can notice that the packages a bit coupled as well. For example, 52 packages (about 65%) depends on 51 other packages (about 64%) which in turn depends on the first ones directly or indirectly.

Every package depends by about other 7 and is depended by about other 7 as well.

This also allows us to see that the main package is net.sourceforge.ganttproject which depends on more packages (more than 60% of those considered).

**Conclusion**

This tables taken from the data generated by ReloadedMetrics allowed us to identify the main package, the main class and that the project is very highly accoupled. We also have seemed that there are a good portion of the interfaces that are fine (not very accoupled) and others that aren’t. Also, apparently there is more decency in classes than there is in interface which is bad.

**Note:** sometimes I referred classes when the right thing to say was interface because of the definitions of the metrics parameters. Also, the tables shown in this report are all in the excel file named dependency.xlsx and the script used to generate them in script.ts.