《面向对象软件度量阈值确定方法研究问题、进展与挑战》附录

**Table A2** Definition and literature sources of object-oriented software metrics——coupling

表A**2** 面向对象软件度量定义和文献来源——耦合性(27个)

|  |  |  |  |
| --- | --- | --- | --- |
| 度量名 | 年份 | 定义 | 文献来源 |
| ***CBO*** | 1994 | ***Coupling between classes.*** 类之间的耦合性. 某个类与它的非继承相关类耦合的数量. 一个类与另一个类耦合是指某个类使用另一类的方法或属性. | Chidamber等[1] |
| ***RFC*** | 1994 | ***Response set for classes.*** 类的响应数据集. 某个类的响应数据集包括该类的方法数M和这些M个方法直接或间接调用方法数. RFC指类的响应数据集中的方法数量. | Chidamber等[1] |
| ***Ca*** | 1994 | ***Afferent Couplings.*** 传入耦合. 该度量表示依赖于所度量类的类数量. | Martin等[2] |
| ***Ce*** | 1994 | ***Efferent Couplings.*** 传出耦合. 该度量表示被度量类所依赖类的数量. | Martin等[2] |
| ***I*** | 1994 | ***Instability.*** 传入和传出耦合下的不稳定性. . 当I=0时,表示最大的稳定性; 当I=1时,表示最大的不稳定性. | Martin等[2] |
| ***MPC*** | 1993 | ***Message passing coupling.*** 消息传递的耦合性. 某一个类中方法调用的数量. | Li等[3] |
| ***DAC*** | 1993 | ***Data abstraction coupling.*** 数据抽象耦合性. 某一个类拥有与另一个类的相同属性类型的数量. | Li等[3] |
| ***ICP*** | 1995 | ***Information-flow-based coupling.*** 信息流的耦合. 某一个类中方法调用的数量, 且以方法调用参数的数量加权后得到. | Lee等[4] |
| ***IH-ICP*** | 1995 | 与ICP类似. 但只计算某个类祖先的若干类的方法调用数量(即基于继承的耦合). | Lee等[4] |
| ***NIH-ICP*** | 1995 | 与ICP类似, 但计算调用与继承类不相关类的方法数. | Lee等[4] |
| ***ACAIC***  ***ACMIC***  ***AMMIC***  ***DCAEC***  ***DCMEC***  ***DMMEC***  ***OCAEC***  ***OCAIC***  ***OCMEC***  ***OCMIC***  ***OMMEC***  ***OMMIC*** | 1997 | 这12个耦合度量是通过类之间的相互作用计算的. 对这些度量的缩写反应相互作用的计算过程:  1. 第一个字母表明相互关系: A表示与祖先类的耦合; D表示与派生类的耦合; O表示与其他类(如没有继承关系)的耦合.  2. 第二个至第三个字母表示相互作用的类型: CA表示类c与类d之间类与属性的交互过程, 即类c中有一个与类d相同类型的属性; CM表示类c与类d之间类与方法的交互过程, 即类c中有一个与类d相同参数类型的方法; MM表示表示类c与类d之间方法与方法的交互过程, 即类c调用类d的一个方法, 或类d的一个方法对类c的一个方法传递参数(函数指针).  3. 最后两个字母表示耦合影响的场所: IC表示内部耦合; EC表示外部耦合. | Briand等[5] [6] |
| ***CBI*** | 1996 | Coupling based on inheritance. 基于继承耦合. 它是某一个类内部所有方法执行的复杂度之和与其派生类数量的乘积. 某个类中的方法复杂度是由霍尔斯特德软件科学工作(Halstead’s software science)测量得到的。 | Kim等[7] |
| ***NAS*** | 1998 | Number of Associations. 类与同级之间的关联数量. 该度量直接从设计文档中收集数据, 采用自上而下的目标—问题度量(GQM)方法[8]收集. 它与CBO相同, 除非以下情况: 如果一个类使用另一个类的方法超过一次, CBO度量对每一次使用作为单独发生的耦合来计数; 而对于NAS, 一个类对另一个类的方法多次调用, 只算一次单独发生的耦合来计数. 因此, 它与CBO有很强的相关性, 但两者并不是正交的. | Harrison等[9] |
| ***DCC*** | 2002 | Direct Class Coupling. 类的直接耦合. 该度量对一个类直接耦合的不同类计数, 包括属性声明中直接相关和方法中信息传递的类. | Bansiya等[10] |
| ***IC*** | 1999 | Inheritance Coupling. 继承耦合数. 该度量表示与某个类耦合的父类数量. 当某个类的继承方法在功能上依赖于该类中新方法或重新定义的方法, 该类将与其父类耦合. 具体来说, 如果有一个继承方法满足以下条件之一, 则称一个类与其父类耦合: 使用一个新方法或重新定义的方法里的属性; 调用重新定义的方法; 被重新定义的方法调用; 和使用在重新定义方法中定义的参数. | Tang等[11] |
| ***CBM*** | 1999 | Coupling Between Methods. 方法间的耦合. 与所有继承方法都耦合的新方法或重新定义的方法总数. 当满足IC度量中至少一条耦合条件时, CBM度量的耦合条件被满足. | Tang等[11] |

**参考文献：**

[1] Chidamber SR, Kemerer CF. A metrics suite for object oriented design. IEEE Transactions on software engineering, 1994,20(6):476-493.[doi: 10.1109/32.295895]

[2] Martin R. OO design quality metrics. An analysis of dependencies, 1994,12(1):151-170

[3] Li W, Henry S. Object-oriented metrics that predict maintainability. Journal of systems and software, 1993,23(2):111-122.[doi: 10.1016/0164-1212(93)90077-B]

[4] Lee Y-s, Liang B. Measuring the coupling and cohesion of an object-oriented program based on information flow. Proceedings of the International Conference on Software Quality, 1995: 81-90

[5] Briand L, Devanbu P, Melo W. An investigation into coupling measures for C++. Proceedings of the 19th international conference on Software engineering, 1997: 412-421.[doi: 10.1145/253228.253367]

[6] Briand LC, Daly JW, Wust JK. A unified framework for coupling measurement in object-oriented systems. IEEE Transactions on software Engineering, 1999,25(1):91-121.[doi: 10.1109/32.748920]

[7] Kim EM, Kusumoto S, Kikuno T, Chang OB. Heuristics for computing attribute values of C++ program complexity metrics. Proceedings of 20th International Computer Software and Applications Conference: COMPSAC'96, 1996: 104-109.[doi: 10.1109/CMPSAC.1996.542433]

[8] Basili VR, Rombach HD. The TAME project: Towards improvement-oriented software environments. IEEE Transactions on software engineering, 1988,14(6):758-773.[doi: 10.1109/32.6156]

[9] Harrison R, Counsell S, Nithi R. Coupling metrics for object-oriented design. Proceedings Fifth International Software Metrics Symposium. Metrics (Cat. No.98TB100262), 1998: 150-157.[10.1109/METRIC.1998.731240]

[10] Bansiya J, Davis CG. A hierarchical model for object-oriented design quality assessment. IEEE Transactions on software engineering, 2002,28(1):4-17.[doi: 10.1109/32.979986]

[11] Tang M-H, Kao M-H, Chen M-H. An empirical study on object-oriented metrics. Proceedings sixth international software metrics symposium (Cat. No. PR00403), 1999: 242-249.[doi: 10.1109/METRIC.1999.809745]