基于构件库管理系统的构件复用度度量模型

薛云皎　王渊峰　余枝强　钱乐秋
（复旦大学计算机科学与工程系软件工程实验室，上海 200433）
E-mail：yunjiaoxue@hotmail.com

摘　要　构件概念及其理论的发展始终以软件复用为切入点，只有被最大可能地复用，构件才有其存在的意义和经济价值。如何用一种较为精确的、基于实践应用的方法来评价构件的复用度，关系到构件质量的提高和构件库的有效性。该文介绍了构件复用的概念，并结合构件库管理系统，提供了一个对构件复用度的度量模型，分析了度量结果对构件库的反馈作用，能够对构件库管理系统的设计与实现提供参考。

关键词　构件　构件库管理系统　度量　复用　检用度

文章编号 1002-8331-(2002)13-0081-04　**文献标识码** A　**中图分类号** TP319

Metrics Model of Component Reuse Degree Based on Component-Base Management System

Xue Yunjiao　Wang Yuanfeng　Yu Zhiqiang　Qian Leqiu
(Dept.of Computer Science and Engineering, Fudan University, Shanghai 200433)

Abstract: The concept and theory of component always lay emphasis on software reuse.A component may achieve its meaning of existing and economic value only after it is reused as possible.It is referred to the improvement of components and efficiency of a component-base that how to evaluate the metrics degree of components with an accurate and practice-based method.This paper introduces the concept of component reuse, provides a metrics model for component reuse degree and analyses its effect on component-base.It can also help the design and implementation of a component-base.

Keywords: Component, Component-base Management System, Metrics, Reuse, Rate of Retrieve and Quote

1　构件技术概述

构件是软件系统中可以明确识别的、可被其他系统复用的成分，可以是需求分析、设计、代码、测试用例、文档或软件开发过程中的其它产品。Bellinzoni 在“Reusing Specifications in OO Applications”中对构件的描述为：

“构件在不同的抽象层次被定义和存储为规约、设计和实现——每个类是来自以前应用的某产品的工程化描述。规约知识——开发知识——被以复用建议(reuse-suggestion)类的形式存储，它们包括对以构件的描述为基础检索可复用构件及检索后组装和剪裁构件的指导。”

在这一描述中体现出了构件描述和检索的的重要意义。要设计、开发和复用一个构件，首先必须对构件进行描述，构件的描述是通过构件模型和构件描述语言来实现的。构件模型是构件的抽象描述，是构件描述语言的基础，模型集中体现了设计者的思想。当前学术界常用的构件模型有 3C、REBOOT、CORBA/OM、OLE/COM、EJB 等。可复用构件是指可被其它系统的开发者复用以开发新软件的构件(通常所述的构件主要是指可复用构件)。软件构件的可复用程度是指软件构件在开发各种软件时可被复用的难易程度。软件构件越具体，其复用程度越低。为了提高软件构件的可复用度，应尽量把构件一般化，使软件开发人员能够有效地复用这些构件及其设计思想。

2　对可复用构件的要求

软件生产的构件化是实现软件复用、提高软件生产率和质量的一个有效方法和趋势。为此，要求软件开发人员将注意力转移到构件的开发上，从事可复用构件的开发。可复用构件的目标在于被广泛地复用，一个构件的复用次数越多，其价值也越大，从一般意义上来讲也具有较高的质量。为使构件能具有较高的可复用性，可复用构件应满足以下条件：

(1)构件的设计应具有较高的抽象程度；

(2)构件应易于调整以适用于具体的环境；

(3)构件应具有良好的封装性和良好定义的接口，易于组装成软件系统；

(4)构件必须具有可检索性，以便开发人员能从构件库中选取到所需构件；

(5)构件必须经过充分的测试，以最大限度地降低错误发生的概率。

3　基于构件库管理系统的构件复用度

3.1　构件库管理系统简介

作者简介：薛云皎，硕士研究生，研究方向：软件工程，构件库管理系统，基于构件、构架的软件开发方法。王渊峰，博士研究生，研究方向：软件工程，构件库管理系统，构件生产与组装。余枝强，硕士研究生，研究方向：软件工程，构件库管理系统，基于构件、构架的软件开发方法。钱乐秋，教授、博士生导师，研究方向：软件工程，构件库管理系统，构件生产与组装，软件测试，基于构件、构架的软件开发方法。

经验与研究表明,软件复用是提高软件生产率、软件质量和降低软件成本的有效方法,而随现代计算机软件理论和技术发展起来的构件技术,作为软件复用概念具体化和软件开发部件化的体现,正受到越来越多研究人员的瞩目,越来越多的研究机构、开发商以及私人已经开发出不计其数的构件。在构件种类和数量达到一定程度之后,自然地对构件的存放和管理提出了更高的要求,需要有一种管理机制来处理大量构件的存储、检索和复杂的相互关系,这与数据量的增长、数据间关系的复杂化和数据库管理系统的出现是类似的过程,于是构件库管理系统的思想便应运而生。目前构件库管理系统的理论尚处于研究阶段,北京大学青鸟项目组已经对构件库概念模型进行了长期的探索,提出了包括功能、使用环境、应用领域、层次、表示方法等在内的刻面模型和管理、检索、度量的框架,为构件库管理系统的研究提供了极好的借鉴。

构件库的关键技术在于构件的描述和检索,它们直接影响到构件库的查准率(Precision)、查全率(Recall)与效率(Efficient)。对构件库的使用者来说,一个构件首先必须能够被检索到,才能考虑复用,因而构件库对构件检索的支持能力直接影响到构件的复用度,复用度与构件的检索和复用形成依赖关系。

3.2 复用度度量模型

度量对于评价构件质量、可复用度等具有重要的意义,度量结果还可指导构件的改进和构件库的管理。通常,度量的困难在于主观因素的参与度大,而且软件度量受到的批评之一就是缺乏坚实的理论基础。

本模型目标在于从客观数据的测量来评价构件复用程度,分析影响因素,为改进构件提供参考。在基于构件复用的软件开发活动结束后,可应用本模型中提出的方法对所使用的可复用构件作出评价,提出修改或改进意见,以提高将来的可复用性。

构件库管理系统应提供必要功能,以支持度量模型从构件库取得如下信息,用作对构件进行新的分析和评价的参考;对构件的分析结果可存入构件库供下一次评价之用:

构件描述信息——提供对构件的基本认识;

构件度量历史数据——对构件的每次度量结果都应在构件库中存储记录;

构件演化历史信息——提供构件评价对构件演化产生的影响的轨迹;

从上一次度量至今构件被检索的次数;

从上一次度量至今构件被引用的次数。

构件库中的构件要得到复用,首先必须能够被检索到,因此构件的复用程度与构件被检索到的情况和被复用的情况都有关系。在侧重于查准率的构件库中,对一个构件进行多次测量时,被检索次数的增加或减少可能反映构件描述的精确度及被检索的情况,但由于用户检索要求的随机性,并不能反映构件的质量。同理,被复用次数也不能反映构件的真实面貌。由于构件复用度同构件被检索到的次数和被复用的次数有关,因此用一段时间内复用次数与检索次数的比值来衡量构件的复用情况势必能较客观地反映构件被复用的程度。

设从上一次评价开始以来,一个构件的被检索次数为 R(即出现在检索结果集中的次数),被引用次数为 Q。R 越大,表明该构件的可检索性越高、抽象程度越高,因而在相关检索中命中的概率更大。Q 越大,表明该构件的可复用性越高,即较好地满足易于调整、易于组装、充分测试的要求,因而被用户引用的概率更高。定义可复用构件的检用率(Rate of Retrieve and Quote)为:

$$RRQ=\frac{Q}{R}$$

上式中 $Q<=R$。在对一个构件的两次测量之间若 $R=0$ 即构件未被检索到,则定义 RRQ 等于上一次测量的 RRQ。RRQ 体现构件被检索与复用的比例关系,可作为复用度的基本参考指标。这里要指出的是,如果构件库的设计侧重于满足查全率,那么大部分构件的被检索次数都会比较多,而被复用次数比较少,RRQ 值始终在低水平徘徊,文章中的分析将不具有普遍意义。

如果 RRQ 极小,则表明构件出现于检索结果的次数远大于被引用次数,可能是描述不精确而经常出现在检索结果中,或因质量不高导致人们较少选用,构件应进行改进;RRQ 适中表明构件已经具有一定的复用市场;RRQ 极大表明该构件描述精确,检索到基本就被复用,质量较好,具有非常高的复用价值。

构件的评价数据必须存入构件库管理系统,从构件的多次评价可以绘出构件的 RRQ 曲线,如图 1 所示:(虚线为测量次数足够多情况下的理想光滑曲线)

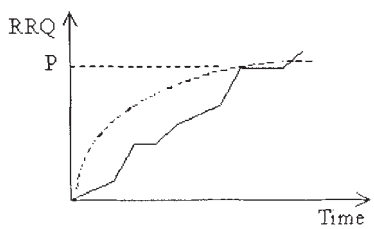

图 1 RRQ 曲线

RRQ 曲线应具有以下性质:

(1)RRQ>=0

(2)Time=0 时,RRQ=0;$\lim RRQ=p$,Time$\to\infty$,$0<p<1$。

(3)f(Time,RRQ)为单调增函数。

如 f(Time,RRQ)为非单调增函数,表明在构件演化过程中产生负效应导致 RRQ 降低,这在构件改进中是应极力避免的倒退。

进一步讨论 由 RRQ 曲线示意图可知:

(1)RRQ 存在一个峰值 P,在到达峰值之前 RRQ 可呈加速度或减速度趋势尽量增长;

(2) 到达峰值 P 之后,RRQ 的增长所需代价过高,以至 RRQ 的增长相对所付出的代价是亏损的和不必要的;

(3)f(Time,RRQ)可能满足不同的分段函数关系式,函数曲线在某点的斜率 $\Delta f/\Delta$Time 代表单位时间内 RRQ 的增长率。RRQ 曲线某一段的斜率越大,表明单位时间内复用次数越频繁,构件质量越高,构件库获得的收益也越大。

前面的讨论都是在构件发生演化的前提下推断 RRQ 曲线的性质,其实践基础在于:在构件描述的精确度和检索方法的精确度达到一定程度的条件下,任何一个构件在结果集中的出现应该是慎重的;由于使用构件需要付费,用户对构件的选择也是慎重的。如果用户对一个构件的检索情况发生次数发生显著变化,那么该构件被引用的次数也应该按相同趋势发生变化,两个数值的比值具有一定的规律性。因而从概率论的角度来看,一个构件的 RRQ 值不会出现随意的变化。如果构件在演

化中可复用性得到提高,则构件出现在结果集中时,被用户选择的可能性应该增大,从而导致 RRQ 值增大。如果构件库和构件本身没有任何变化,那么在相连两次测量中,一个构件的 RRQ 值应在一定范围内可视为保持不变(例如:变化程度不超过 10%)。

为体现在构件库和构件保持不变的条件下 RRQ 值的不变性,设计了一个获得度量数据的方法:假设构件库中含有 N 个构件,在一次基于查准率的检索中得到 n 个构件,从中选择一个构件,则查到的 n 个构件的 R 值都加 1,而仅有被选择的那个构件的 Q 值加 1。因为检索要求的出现是随机的,在不考虑构件性能的条件下,对构件的选取也可视为随机的,因而算法可以描述为:对编号为 1 到 N 的构件,每次取 n 个随机数 $c1$,$c2$,...cn,代表编号为 $c1$,$c2$,...cn 的构件被检索到,$R(c1,c2,...cn)=R(c1,c2,...cn)+1$;再从 1 到 n 之间取一个随机数 m,代表编号为 m 的构件被选用,$Q(m)=Q(m)+1$。重复若干次,得到两个数组 $R(1,...,N)$和 $Q(1,...,N)$。用 $Q(i)$除以 $R(i)$就得到另一个数组 RRQ(1,...,N)。上述过程重复多次,则可以得到多次测量的 RRQ,绘出多个构件的 RRQ 曲线如图 2 所示:

可见,大部分构件的 RRQ 都在一定范围内保持不变。

如果给构件加上权重,每次检索到的 n 个构件不是以均等的可能性被选中,而按照权重高低获得不同的选取机会,得到的 RRQ 曲线如图 3 所示:

RRQ 按照构件权重大小分成了几组,每个组中也基本保持在一定范围内不变。在实际的检索过程中,由于用户的检索要求具有目的性,对结果选择具有意识性,在排除随机因素的影响后,构件的 RRQ 更能显出规律性和稳定性。

在实际应用中,RRQ 曲线的变化趋势可以指导构件库管理者发现问题:如果 RRQ 曲线持续呈下降趋势,原因可能是 R 增大超过 Q 的增大,意味着构件的描述可能有问题,并且很可能是新入库的构件与该构件描述过于相似,以至该构件经常和新构件共同出现在结果集中,但用户总是选择另一个构件。原因也可能是 Q 减少超过 R 减小,意味着用户对构件的接受程

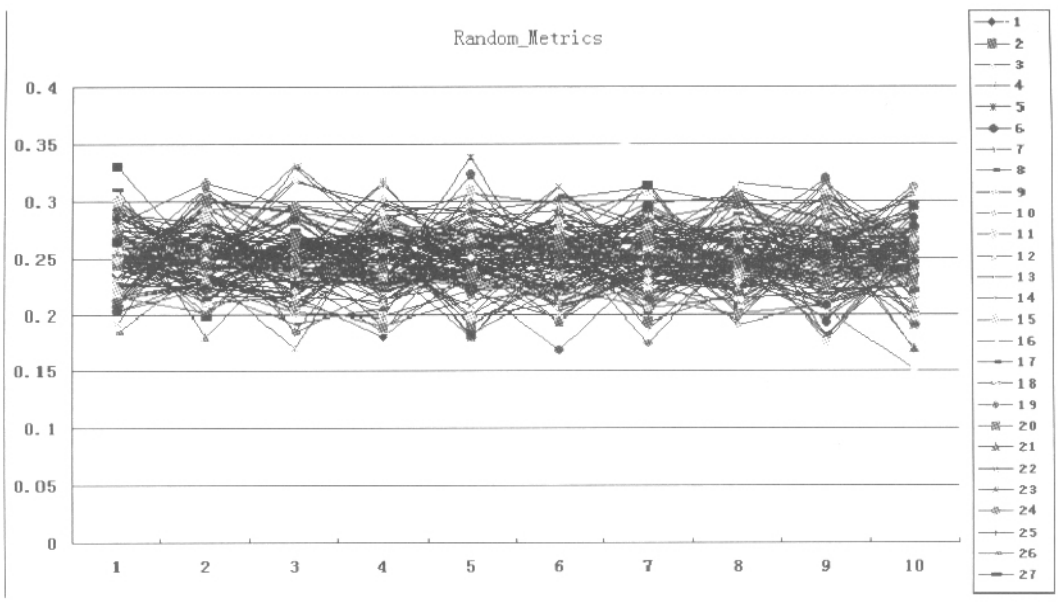

图 2

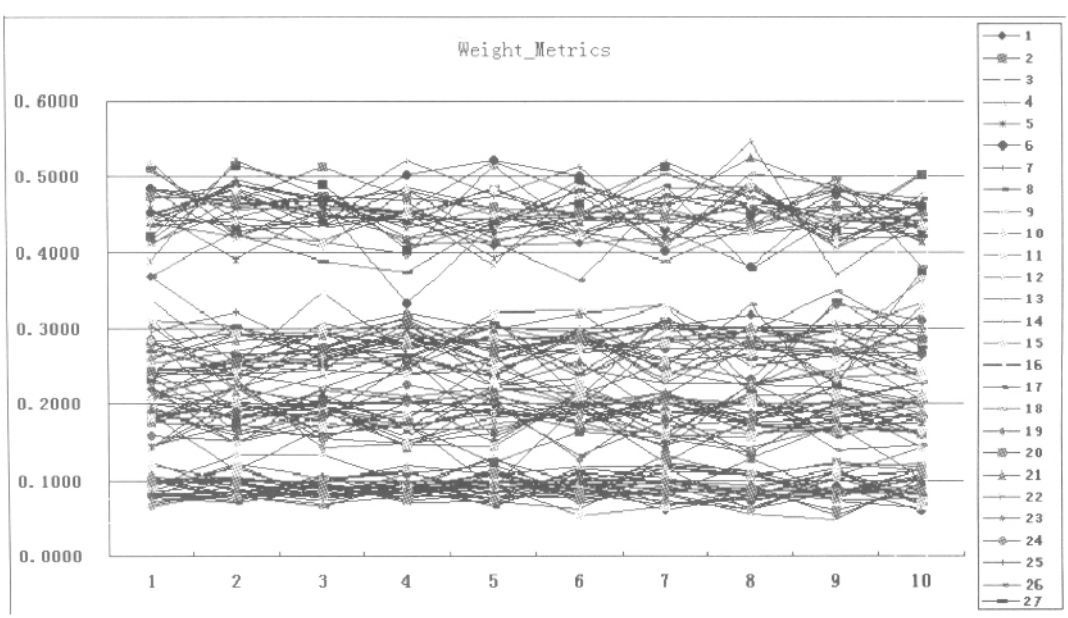

图 3

度降低,构件是否应该改进?或者是构件的生命周期已经接近尾声?如果 RRQ 持续呈上升趋势,结果总是乐观的,表明构件的演化是朝提高性能的方向进行。

4 结语

构件库管理系统中构件的描述机制是构件检索的基础,而检索算法又直接制约检索的有效性。在构件可检索度达到一定程度的前提下才能考虑构件的被复用的程度,二者结合成为评价基于构件库管理系统的构件复用度的两个关键方面。该文以 RRQ 曲线为核心分析了构件的检用率,并在实验数据的基础上分析了构件 RRQ 的变化规律及对分析构件复用情况的帮助,能够对构件及构件库管理系统的研究和开发提供一定的参考。尽管构件的复用度与构件质量具有一定的正向比例关系,但在将来的工作中,还应该在软件质量评估的基础上提出对构件质量的评价模型,更精确地为构件提供评价。

(收稿日期:2001 年 7 月)

参考文献

1.Pedro Esteves Pinto.Promoting Software Reuse in a Corporate Setting [M].School of Computer Science,Carnegie Mellon University

2.Bellinzona R,M G Gugini,B Pernici.Reusing Specifications in OO Applications[S].IEEE Software,1995

3.Roger S Pressman.Software Engineering,A Practitioner's Approach[M]. Fourth Edition.McGraw-Hill Press,1997

4.Ivar Jacobson,Martin Griss,Patrik Jonsson.Software Reuse[M].1997

5.梅宏,谢涛,袁望洪等.青鸟构件库的构件度量[J].软件学报,2000;11(5):634~641

6.汪洋.基于软件构架和构件的软件演化研究[D].硕士毕业论文.复旦大学

(上接 61 页)

文献[11]给出了 RSA 和基于普通素数域上离散对数问题的 ElGamal 加密体制的软件(包括加解密和数字签名)的实现速度,其实验数据是在 SPARC II 上运行得到的。文献[7]给出了椭圆曲线上数字签名软件的实现速度,其实验数据是在 Pentium Pro 200MHz PC 上得到的。由于 RSA 和 ECES 中的基本运算存在差异,很难对它们的运算量进行仔细的比较,下面采用 RSA 公司论坛中一种比较粗略等价进行评价,即:一次椭圆曲线上点的加法大致相当于 RSA 中的 10 次模乘运算[10],表 2 列出了这种粗略的比较:

表 2 椭圆曲线加密系统(160 比特),1024 比特的 RSA 和传统有限域上离散对数加密系统的性能比较

	GF(2^{160})上的 ECDSA 或 ECES	RSA,n 为 1024 比特,e=216+1,	基于 1024 比特素数的离散对数加密系统
加密	120	17	480
解密	60	384	240
签名	60	384	240
校验	120	17	480

表 2 中的数字为完成操作所需的时间单位数,其中一个时间单位定义为完成一次 1024 比特的模乘所需要的时间。此表只是进行粗略的比较,并没有考虑可能的优化。通过比较发现,尽管 RSA 的密钥比较长,但是它对签名的验证上还是具有优势的。

6 结束语

通过上面的比较分析,显然基于椭圆曲线的加密系统具有非常光明的前景,相对 RSA 来讲它具有如下优点:

(1)在相同安全性的基础上,椭圆曲线的加密系统密钥长度更短,这在存储量少的应用中是非常重要的,IC 卡是典型的小存储量的设备,而且在金融系统的身份认证中有着广泛的应用,短密钥为这种应用提供了方便。短密钥的另外一个好处是密文和数字签名比较短,在对短信息(比如小于 1024 比特的信息)进行数字签名时具有优势。

(2)总体来讲,基于椭圆曲线的加密系统的运算更快,在网络传输速度越来越快的情况下,缓慢的加密往往成为一个系统的瓶颈,基于椭圆曲线的加密系统使得这种情形得到一定的改善。

基于其自身的优点,椭圆曲线密码学一出现便受到关注。现在密码学界普遍认为它将代替 RSA 成为通用的公钥密码算法,SET 协议的定制者已经把它作为下一代 SET 协议中缺省的公钥密法算法,目前椭圆曲线密法学正是密码学界研究的热点,是很有前途的方向。(收稿日期:2002 年 3 月)

参考文献

1.W Diffie,M E Hellman.New Directions in Cryptography[J].IEEE Transactions on Information Theory,1976;IT-22(6):644~654

2.R L Rivest,A Shamir,L M Adleman.A Method for Obtaining Digital Signatures and Public-Key Cryptosystems[J].Communications of the ACM,1978;21(2):120~126

3.A K Lenstra,H W Lenstra,Jr et al.Lecture Notes in Mathematics 1554:The Development of the Number Field Sieve[M].Springer-Verlag,1993

4.T ElGamal.A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms[J].IEEE Transactions on Information Theory,1985;IT-31(4):460~472

5.Standard Specifications for Public Key Cryptography[S].IEEE P1363/D13(Draft Version 13),1999

6.A J Menezes.Elliptic Curve Public Key Cryptosystems[M].USA:Kluwer Academic Publishers,1993

7.Naoya Torii,Kazuhiro Yokoyama.Elliptic Curve Cryptosystem[J].FUJITSU Sci Tech J,36(2):140~146

8.V Miller.uses of elliptic curves in cryptography[M].Advances in Cryptology-CRYPTO ´85,Lecture-Notes in Computer Science,Springer-Verlag,1990;435:218~238

9.A Menezes.Elliptic Curve Public Key Cryptosystems[M].Kluwer Academic Publshers,1993

10.M J B Robshaw,Yiqun Lisa Yin.Elliptic Curve Cryptosystems.An RSA Laboratories Technical Note,Revised,1997:URL-http://www.rsa.com/rsalabs/ecc/elliptic_curve.html

11.J B Lagarias,D P Mitchell,W M Schell.CryptoLib:Cryptography in Software[C].In:UNIX Security Symposium IV Proceedings,USENIX Association,1993:1~17

12.吴世忠,祝世雄等.应用密码学-协议、算法与 C 源程序[M].机械工业出版社,2000