
并发面向对象中的继承反常现象	961
Inheritance Anomaly in Concurrent Object Orientation *	961
1 认识继承反常现象	962
2 理解继承反常现象	962
2.1 区别 Inheritance 和 Subtyping	962
2.2 渐增式继承	962
2.2.1 定义继承反常现象	961
2.2.2 解释继承反常现象	961
3 结束语	961
References:	961
附中文参考文献:	961

并发面向对象中的继承反常现象*

王生原¹⁺, 杨良怀², 袁崇义³, 杨 萍⁴

¹(清华大学 计算机科学与技术系, 北京 100084)

²(新加坡国立大学 计算学院 计算机科学系, 新加坡)

³(北京大学 计算机科学技术系, 北京 100871)

⁴(兰州大学 信息科学与工程学院, 甘肃 兰州 730000)

Inheritance Anomaly in Concurrent Object Orientation*

WANG Sheng-Yuan¹⁺, YANG Liang-Huai², YUAN Chong-Yi³, YANG Ping⁴

¹(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

²(Department of Computer Science, School of Computing, National University of Singapore, Singapore)

³(Department of Computer Science and Technology, Beijing University, Beijing 100871, China)

⁴(School of Information Science and Technology, Lanzhou University, Lanzhou 730000, China)

+ Corresponding author: Phn: +86-10-62645407, Fax: +86-10-62645000, E-mail: wsy@163.net, <http://www.tsinghua.edu.cn>

Received 2002-10-08; Accepted 2003-01-05

Abstract: The combination of concurrency and object orientation is definitely natural except for inheritance. One of the interferences between inheritance and concurrency is inheritance anomaly. Although having been researched extensively, inheritance anomalies are still only vaguely defined and often misunderstood, and no much formal work has been done. A new viewpoint is set forth for understanding inheritance anomalies, in which each subtyping relation has its specific incremental inheritance. Related concepts and definitions are formalized through the language of Category. Some issues are well adapted to distinguish and explain different standpoints about inheritance anomalies, and can serve as guidelines in the modeling of inheritance.

Key words: concurrency; object orientation; inheritance anomaly; incremental inheritance; category theory

* Supported by the National Natural Science Foundation of China under Grant No.60373000 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant Nos.863-317-01-04-99, 2001AA115126 (国家高技术研究发展计划(863)); the National Grand Fundamental Research 973 Program of China under Grant No.G1999032706 (国家重点基础研究发展规划(973)); the Key Science-Technology Project of the National 'Ninth Five-Year-Plan' of China under Grant No.98-780-01-07-03 (国家“九五”重点科技攻关项目); the Defense Pre-Research Project of the 'Ninth Five-Year-Plan' of China (国家“九五”国防预研基金); the National Research Foundation for the Doctoral Program of Higher Education of China (国家教育部博士点基金); the Ministry & Commission-Level Research Foundation of China (部委级基金); the Natural Science Foundation of Gansu Province of China under Grant No.ZS991-A25-014-G (甘肃省自然科学基金)

作者简介: 王生原(1964—),男,山西应县人,博士,副教授,主要研究领域为分布对象计算,Petri 网应用,嵌入式软件环境;杨良怀(1967—),男,博士,讲师,主要研究领域为数据库系统实现技术,数据仓库,半结构化数据,Web 数据集成,数据挖掘;袁崇义(1941—),男,教授,博士生导师,主要研究领域为并行计算,Petri 网理论及应用;杨萍(1964—),女,副教授,主要研究领域为人工智能。

摘要: 如果不考虑继承性,并发性与对象技术的结合是很自然的.继承反常(又称继承异常)现象是继承性和并发性不相容的主要原因之一.现阶段人们对继承反常现象的认识有许多模糊之处,出发点不尽相同,形式化的工作也很少.对不同的 subtyping 关系考虑其特有的渐增式继承方法有利于把握继承反常现象的实质,也丰富了“在并发面向对象语言中应将 inheritance 层次和 subtyping 层次区别对待”这一认识的内涵.在阐述基本观点之后,采用范畴论的术语对相关的概念和定义做了形式化工作.一些观点和结论适用于区分和解释相关工作的出发点和贡献,并对并发面向对象技术中继承性的建模问题有所启示.

关键词: 并发性;面向对象;继承反常;渐增式继承;范畴论

中图法分类号: TP301

文献标识码: A

Inheritance(继承)是类(class)之间的一种层次关系.在一般的面向对象语言中,类层次自动对应了一种类型(type)层次^[1],这样,superclass 和 subclass 之间的关系自然是一种 supertype 和 subtype 之间的关系.

1 认识继承反常现象

如图1所示,类Buffer实现了一个有界缓冲区类型,可以并发地接受消息 put 和 get.同步机制采用了“method guards”方式,即为每一个 method 附加一个 guard 谓词^[2].构造 Buffer 的另一个子类 LockableBuffer 也会引发同样的问题.这种“为获得有效继承而必须对父类代码进行实质性修改的现象”就是所谓的继承反常(inheritance anomaly)^[3~5].

2 理解继承反常现象

2.1 区别Inheritance和Subtyping

综合各种观点,要理解和解释继承反常现象,首先必须正确区别 inheritance 层次和 subtyping 层次,并深刻领会二者的联系^[6,7].对于这个问题,文献[1,8]的观点是比较权威的.在此基础上,我们的理解是:Inheritance 是在代码层次上作修改,而 subtyping 是在语义层次上作修改.前者是代码共享的一种重要途径,但不能保证 subclass 能够继承 superclass 的行为;后者要求 subtype 保持 supertype 的某种外部可观察行为(或语义行为),在规范一级共享,同代码没有关系.Inheritance 层次关系可以理解为“is_similar_to”(或“like”)的关系,而将“is_a”关系更适合用在理解 subtyping 层次关系上,见表1^[9].

命题 1. 设 $R \subseteq R_L, R' \subseteq R, R$ 定义的 Class 范畴为 C_R, R' 定义的 Class 范畴为 $C_{R'}$, 则 $C_{R'}$ 是 C_R 的子范畴, 并称之为 C_R 的子 Class 范畴.

证明: 对任何 $A, B \in ob \quad C_R = ob \quad C_{R'}, hom_{C_{R'}}(A, B) \subseteq hom_{C_R}(A, B)$.

推论 1. 任何 $R \subseteq R_L$ 定义的 Class 范畴 C_R 都是 C_L 的子 Class 范畴.

Table 1 Latency of macro-instructions

表 1 宏指令的时延

Acronym	Latency (ns)		Event symbol
	Value	Symbol	
NSI	87	τ_α	$n(\alpha, e)$
NST	154	—	—
NSE	904 (1.4%)	τ_γ	$n(\gamma, e)$
NSO	150	τ_o	$n(o, e)$
NSA	303	τ_β	$n(\beta, e)$

2.2 渐增式继承

Subtyping 要求 subtype 保持 supertype 的某种行为(可看作是一种不变量,比如同步约束)^[10].Subclass 在增加新的属性或方法时,为了避免破坏这种不变量,难免要对继承的代码进行扩展或修改.这种扩展或修改很可能是重大的或实质性的,结果使得代码共享失去意义.这便是继承反常的直观含义.

假定非线性优化问题:

$$\min_{z \in X} \iint_{\Omega} \sqrt{1 + z_x^2 + z_y^2} dx dy \quad (1)$$

的近似解为 $z(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 b_{ij} B_i^n(x) B_j^n(y)$, 其中

$b_{0j}, b_{3j}, b_{10}, b_{13}$ 由边界条件决定.确定其余几个 Bézier 系数是一个非线性约束优化问题.我们采用基于一维线性搜索的整体收敛的 Newton 迭代方法来求解.其基本的迭代格式为

$$\mathbf{X}^{i+1} = \mathbf{X}^i + \lambda_i \cdot \mathbf{p}_n.$$

步长 λ_i 采用回溯线性搜索方法确定,下降方向 $\mathbf{p}_n$ 是 Newton 方向,这种非线性优化方法是一个比较成功的具有整体收敛性的方法,对于初值的选择没有苛刻的要求.

基于 TB-CCRD 的新系统采用与 Linux 直接路由式虚拟服务器相同的框架结构^[11].如图 2 所示,DWSS 的各个后端服务器通过高速以太网相互连接,它们屏蔽 ARP 协议,并拥有与前端机相同的 IP(记为 vIP)和 Web 服务端口号(记为 vPort),在用户看来,这个系统就相当于一个 IP 地址为 vIP,服务端口号为 vPort 的 Web 服务器.在新系统中,前端机负责接收来自用户的数据包,发放用于指示处理该数据包最适当的服务器 ID(可以是后端机在系统内部的标识符,也可以是它的 MAC 地址)的标记,并转发数据包;后端机则负责处理由前端机转发而来的用户数据包并直接回应用户,具体包括建立/拆除与用户的 TCP 连接、TCP 连接转交以及提供 URL 内容等工作.

2.2.1 定义继承反常现象

本节是对上述观点的形式化描述,可与第 2.2 节对照阅读.范畴论的观点层次较高,易于抽象出问题的本质.

与程序 P 在 F 下运行等价的错误影响程序 P_F 有

2.2.2 解释继承反常现象

例 1:在谈到顺序面向对象语言时,一般不涉及继承反常现象.这是因为,这类语言 L 都默认一个特殊的 subtyping 关系 p (由 R_L 定义的完整 inheritance 层次关系被默认为相应于 p 的渐增式继承关系),它对应的 Type 范畴记为 T_L ,满足: C_L 实现 T_L ,实现函子为 $F: \forall A(A \in ob C_L \rightarrow FA = type(A)).L$ 无 p -继承反常.因为顺序面向对象语言不提供定义 subtyping 关系

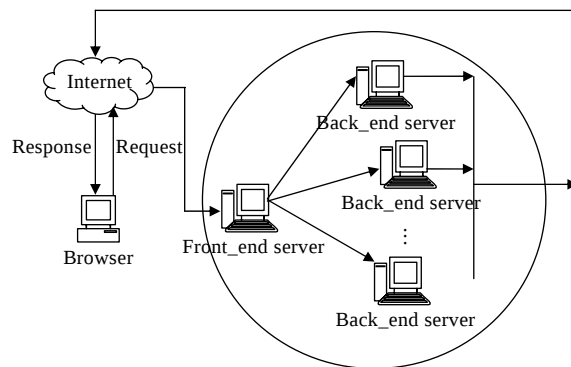


Fig.2 The framework of the TB-CCRD based DWSS

图 2 基于标记的缓存协作 DWSS 系统框架的机制,所以这种默认的 subtyping 关系是语言中唯一的 subtyping 关系,这些语言 L 无继承反常.

3 结束语

本文形式地给出了“继承反常现象”的一种一般性定义.定义对每一种 Subtyping 关系都有其相对应的“渐增式继承”,使其更具有普适性.文中“渐增式继承”是一个相对的概念,这有助于对人们“使继承反常现象得到缓解”的努力得以分类和评价.

致谢 在此,我们向对本文的工作给予支持和建议的同行,尤其是北京大学计算机科学技术系袁崇义教授、屈婉玲教授领导的讨论班上的同学和老师表示感谢.

References:

- [1] Clifton C, Leavens GT, Chambers C, Millstein T. MultiJava: modular open classes and symmetric multiple dispatch for Java. ACM SIGPLAN Notices, 2000,35(10):130-145.
- [2] Wegner P, Zdonik SB. Inheritance as an incremental modification mechanism or what like is and isn't like. In: Gjessing S, Nygaard K, eds. Proc. of the ECOOP'88. LNCS 322, Heidelberg: Springer-Verlag, 1988. 55-77.
- [3] Waxman BM. Routing of multipoint connections. IEEE Journal on Selected Areas in Communications, 1988,6(9):1617-1622.
- [4] Yonezawa A. ABCL: An Object-Oriented Concurrent System. Cambridge: MIT Press, 1990.
- [5] Matsuoka S, Yonezawa A. Analysis of inheritance anomaly in object-oriented concurrent programming languages. In: Agha G, Wegner P, Yonezawa A, eds. Research Directions in Concurrent Object-Oriented Programming. Cambridge: MIT Press, 1993. 107-150.
- [6] Hemige V. Object-Oriented design of the groupware layer for the ecosystem information system [MS. Thesis]. University of Montana, 1995.
- [7] Rose A, Perez M, Clements P. Modechart toolset user's guide. Technical Report, NML/MRL/5540-94-7427, Austin: University of Texas at Austin, 1994.

- [8] Keene SE. A Programmer's Guide to Object-Oriented Programming in Common LISP. Boston: Addison-Wesley Longman Publishing Co., Inc., 1988.
- [9] Guo L, Tang ZS. Specification and verification of the triple-modular redundancy fault-tolerant system. Journal of Software, 2003,14(1):28~35 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/28.htm>
- [10] Schutze, H. Dimensions of meaning. In: Whitelock P, ed. Proc. of the Supercomputing'92. Los Alamitos, 1992. 787~796. <ftp://parcftp.parc.xerox.com/pub/qca/papers/>
- [11] Wang XW. Research on quality-of-service management and group communication mechanisms in distributed multimedia systems [Ph.D. Thesis]. Shenyang: Northeastern University, 1998 (in Chinese with English abstract).

附中文参考文献:

- [9] 郭亮,唐稚松.三机冗余容错系统的描述和验证.软件学报,2003,14(1):28-35. <http://www.jos.org.cn/1000-9825/14/28.htm>
- [11] 王兴伟.分布式多媒体系统服务质量管理与组通信机制的研究[博士学位论文].沈阳:东北大学,1998.