Dovland, Johan; **Johnsen, Einar Broch**; **Owe, Olaf**; **Steffen, Martin**
Lazy behavioral subtyping. (English) Zbl 1204.68072
J. Log. Algebr. Program. 79, No. 7, 578-607 (2010).

Summary: Inheritance combined with late binding allows flexible code reuse but complicates formal reasoning significantly, as a method call's receiver class is not statically known. This is especially true when programs are incrementally developed by extending class hierarchies. This paper develops a novel method to reason about late bound method calls. In contrast to traditional behavioral subtyping, reverification of method specifications is avoided without restricting method overriding to fully behavior-preserving redefinition. The approach ensures that when analyzing the methods of a class, it suffices to consider that class and its superclasses. Thus, the full class hierarchy is not needed, and incremental reasoning is supported. We formalize this approach as a calculus which lazily imposes context-dependent subtyping constraints on method definitions. The calculus ensures that all method specifications required by late bound calls remain satisfied when new classes extend a class hierarchy. The calculus does not depend on a specific program logic, but the examples in the paper use a Hoare style proof system. We show soundness of the analysis method. The paper finally demonstrates how lazy behavioral subtyping can be combined with interface specifications to produce an incremental and modular reasoning system for object-oriented class hierarchies.

MSC:

68N99 Theory of software

Keywords:

object orientation; inheritance; code reuse; late binding; proof systems; method redefinition; incremental reasoning; behavioral subtyping

Software:

Creol; Featherweight Java; JML; Simula 67; Spec#

Full Text: DOI

References:

[1] Abadi, M.; Leino, K.R.M., A logic of object-oriented programs, (), 11-41 · Zbl 1274.68055

[2] Alagic, S.; Kouznetsova, S., Behavioral compatibility of self-typed theories, (), 585-608 · Zbl 1049.68744

[3] America, P., A behavioural approach to subtyping in object-oriented programming languages, (), 173-190 · Zbl 0681.68010

[4] America, P., Designing an object-oriented programming language with behavioural subtyping, (), 60-90

[5] Apt, K.R., Ten years of hoare's logic: a survey — part I, ACM trans. program. lang. syst., 3, 4, 431-483, (1981) · Zbl 0471.68006

[6] Apt, K.R.; Olderog, E.-R., Verification of sequential and concurrent systems, Texts and monographs in computer science, (1991), Springer

[7] Barnett, M.; Leino, K.R.M.; Schulte, W., The spec# programming system: an overview, (), 49-69

[8] ()

[9] Broy, M.; Stølen, K., Specification and development of interactive systems, Monographs in computer science, (2001), Springer

[10] Burdy, L.; Cheon, Y.; Cok, D.R.; Ernst, M.; Kiniry, J.; Leavens, G.T.; Leino, K.R.M.; Poll, E., An overview of JML tools and applications, (), 75-91

[11] Chin, W.-N.; David, C.; Nguyen, H.-H.; Qin, S., Enhancing modular OO verification with separation logic, (), 87-99 · Zbl 1295.68082

[12] C. Clifton, G.T. Leavens, Obliviousness, modular reasoning, and the behavioral subtyping analogy, in: Proc. of AOSD 2003 Wksh. on Software Engineering Properties of Languages for Aspect Technologies, SPLAT 2003 (Boston, MA, March 2003), 2003. Available from: <http://www.daimi.au.dk/⊠eernst/splat03/papers.html>.

[13] Dahl, O.-J., Verifiable programming, Int. series in computer science, (1992), Prentice Hall

[14] O.-J. Dahl, B. Myhrhaug, K. Nygaard, (Simula 67) common base language, Technical Report S-2, Norsk Regnesentral, Oslo, 1968.

[15] de Boer, F.S., WP-calculus for OO, (), 135-149

[16] de Boer, F.S.; Clarke, D.; Johnsen, E.B., A complete guide to the future, (), 316-330

[17] Dhara, K.K.; Leavens, G.T., Weak behavioral subtyping for types with mutable objects, (), 91-113

[18] Dhara, K.K.; Leavens, G.T., Forcing behavioural subtyping through specification inheritance, (), 258-267

[19] Dovland, J.; Johnsen, E.B.; Owe, O., Observable behavior of dynamic systems: component reasoning for concurrent objects, (), 19-34 · Zbl 1277.68056

[20] Dovland, J.; Johnsen, E.B.; Owe, O.; Steffen, M., Lazy behavioral subtyping, (), 52-67

[21] Dovland, J.; Johnsen, E.B.; Owe, O.; Steffen, M., Incremental reasoning for multiple inheritance, (), 215-230 · Zbl 1211.68084

[22] Findler, R.B.; Felleisen, M., Contract soundness for object-oriented languages, (), 1-15

[23] Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J., Design patterns: elements of reusable object-oriented software, (1995), Addison-Wesley

[24] Hoare, C.A.R., An axiomatic basis of computer programming, Commun. ACM, 12, 10, 576-580, (1969) · Zbl 0179.23105

[25] Hoare, C.A.R., Procedures and parameters: an axiomatic approach, (), 102-116 · Zbl 0221.68020

[26] Hoare, C.A.R., Communicating sequential processes, Int. series in computer science, (1985), Prentice Hall · Zbl 0637.68007

[27] M. Huisman, Java Program Verification in Higher-Order Logic with PVS and Isabelle, PhD thesis, Kathol. Univ. Nijmegen, 2001.

[28] Igarashi, A.; Pierce, B.C.; Wadler, P., Featherweight Java: a minimal core calculus for Java and GJ, ACM trans. on program. lang. and syst., 23, 3, 396-450, (2001)

[29] Jacobs, B.; Poll, E., A logic for the Java modelling language JML, (), 284-299 · Zbl 0977.68588

[30] Johnsen, E.B.; Owe, O., Inheritance in the presence of asynchronous method calls, (), paper 282c

[31] Johnsen, E.B.; Owe, O., An asynchronous communication model for distributed concurrent objects, Software and systems modeling, 6, 1, 35-58, (2007)

[32] Johnsen, E.B.; Owe, O.; Yu, I.C., Creol: A type-safe object-oriented model for distributed concurrent systems, Theor. comput. sci., 365, 1-2, 23-66, (2006) · Zbl 1118.68031

[33] Khatchadourian, R.; Dovland, J.; Soundarajan, N., Enforcing behavioral constraints in evolving aspect-oriented programs, (), 19-28

[34] Leavens, G.T.; Wheil, W.E., Reasoning about object-oriented programs that use subtypes, (), 212-223

[35] G.T. Leavens, D.A. Naumann. Behavioral subtyping, specification inheritance, and modular reasoning, Technical Report 06-20a, Dept. of Comput. Science, Iowa State Univ., 2006.

[36] Leavens, G.T.; Leino, K.R.M.; Müller, P., Specification and verification challenges for sequential object-oriented programs, Formal aspects of computing, 19, 2, 159-189, (2007) · Zbl 1121.68074

[37] Liskov, B., Data abstraction & hierarchy, (), 17-34

[38] Liskov, B.H.; Wing, J.M., A behavioral notion of subtyping, ACM trans. on program. lang. and syst., 16, 6, 1811-1841, (1994)

[39] Luo, C.; Qin, S., Separation logic for multiple inheritance, (), 27-40 · Zbl 1286.68320

[40] Mikhajlov, L.; Sekerinski, E., A study of the fragile base class problem, (), 355-382

[41] Owicki, S.; Gries, D., An axiomatic proof technique for parallel programs I, Acta inform., 6, 4, 319-340, (1976) · Zbl 0312.68011

[42] Parkinson, M.J.; Bierman, G.M., Separation logic, abstraction, and inheritance, (), 75-86 · Zbl 1295.68091

[43] Pierik, C.; de Boer, F.S., A proof outline logic for object-oriented programming, Theor. comput. sci., 343, 3, 413-442, (2005) · Zbl 1077.68018

[44] C. Pierik, F.S. de Boer, On behavioral subtyping and completeness, in: Proc. of ECOOP 2005 Wksh. on Formal Techniques for Java-like Programs, FTfJP 2005 (Glasgow, July 2005), 2005. Available from: <http://www.cs.ru.nl/ftfjp/2005.html>.

[45] Poetzsch-Heffter, A.; Müller, P., A programming logic for sequential Java, (), 162-176

[46] Soundarajan, N.; Fridella, S., Inheritance: from code reuse to reasoning reuse, (), 206-215

[47] Soundararajan, N., Axiomatic semantics of communicating sequential processes, ACM trans. on program. lang. and syst., 6, 4, 647-662, (1984) · Zbl 0542.68013

[48] von Oheimb, D.; Nipkow, T., Hoare logic for nanojava: auxiliary variables, side effects, and virtual methods revisited, (), 89-105 · Zbl 1064.68543

[49] D. von Oheimb, Analysing Java in Isabelle/HOL: Formalization, Type Safety, and Hoare-Logics, PhD thesis, Technische Univ. München, 2001. · Zbl 0997.68019

[50] Wehrheim, H., Behavioral subtyping relations for active objects, Formal methods in system design, 23, 2, 143-170, (2003) · Zbl 1057.68015

[51] Wills, A., Specification in fresco, (), 127-135

paper as accurately as possible without claiming the completeness or perfect precision of the matching.