Certification of Quantitative Properties of Programs

Martin Hofmann
Ludwig-Maximilians-Universität, München, Germany

In the context of mobile and global computing knowledge of quantitative properties of programs is particularly important. Here are some typical scenarios:

- A provider of distributed computational power may only be willing to offer this service upon receiving dependable guarantees about the required resource consumption.
- A user of a handheld device, wearable computer, or smart card might want to know that a downloaded application will definitely run within the limited amount of memory available.
- Third-party software updates for mobile phones, household appliances, or car electronics should come with a guarantee not to set system parameters beyond manufacturer-specified safe limits.
- Requiring certificates of specified resource consumption will also help to prevent mobile agents from performing denial of service attacks using bona fide host environments as a portal.

The lectures will describe how such quantitative resource-related properties can be inferred automatically using type systems and how the results of such analysis can be turned into unforgeable certificates using a proof-carrying code framework.

The underlying programming language will be Java, Java Bytecode, and sometimes a functional language translated into Java Bytecode. The properties we will focus on mostly are space usage and values of parameters to system calls.

Much of the material is based on results of the recently completed EU-funded project "Mobile Resource Guarantees" IST-2001-33149 [1,2,4]. However, I will also describe important related work, in particular the analysis by Cachera et al. (not yet referenced), by Unnikrishnan et al [5], and by Mok and Yu [6].

In guided exercises there will be opportunity to experiment with prototype implementations.

References

1. D. Aspinall, St. Gilmore, M. Hofmann, D. Sannella and I. Stark. *Mobile Resource Guarantees for Smart Devices*, In: Construction and Analysis of Safe, Secure, and Interoperable Smart Devices. Proc. of CASSIS 2004, Springer, pp: 1-26, 2005
2. L. Beringer, M. Hofmann, A. Momigliano and O. Shkaravska. *Automatic Certification of Heap Consumption*, In: Logic for Programming, Artificial Intelligence, and Reasoning, Proc. of the LPAR 2004, Springer, pp: 347-362, 2005
3. K. MacKenzie, N. Wolverson. *Camelot and Grail: Resource-aware Functional Programming on the JVM*, In: rends in Functional Programming, Intellect, Vol. 4, pp: 29-46, 2004
4. M. Hofmann, S. Jost. *Static Prediction of Heap Space Usage for First-Order Functional Programs.* In: Proc. of the 30th ACM Symposium on Principles of Programming Languages, ACM Press, Vol. 38, No. 1, pp: 185-197, 2003
5. L. Unnikrishnan, S.D. Stoller and Y.A. Liu. *Optimized Live Heap Bound Analysis*, In: Proc. of the VMCAI 2003, LNCS 2575, Springer, 2003
6. W. Yu, A.K. Mok. *Formal Specification and Verification of Resource Bound Security Using PVS*, In: Proc. of International Symposium on Software Security 2003, pp. 113-133, LNCS 3233, Springer, 2004