Electronic Communications of the EASST
Volume X (2014)

Proceedings of the Automated Verification of Critical Systems (AVoCS 2013)

Physical Type Tracking through Minimal Source-Code Annotation

Dave Donaghy and Tom Crick

1 pages

Guest Editors: Helen Treharne, Steve Schneider
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Physical Type Tracking through Minimal Source-Code Annotation

Dave Donaghy[1] **and Tom Crick**[2]

[1] dave.donaghy@hp.com
HP Bristol, UK

[2] tcrick@cardiffmet.ac.uk
Department of Computing
Cardiff Metropolitan University, UK

Abstract: One of many common artefacts of complex software systems that often needs to be tracked through the entirety of the software system is the underlying type to which numerical variables refer.

Commonly-used languages used in business provide complex mechanisms through which general objects are associated to a given type: for example, the *class* (and *template*) mechanisms in Python (and C++) are extremely rich mechanisms for the construction of types with almost entirely aribtrary associated operation sets.

However, one often deals with software objects that ultimately represent numerical entities corresponding to real-world measurements, even through very standardised SI units: metres per second, kilogram metres per second-squared, etc. In such situations, one can be left with insufficient and ineffective type-checking: for example, the C *double* type will not precent the erroneous addition of values representing speed (with SI units *metre per second*) to values representing mass (SI unit *kilogram*).

We present an addition to the C language, defined through the existing *attribute* mechanism, that allows automatic control of physical types at compile-time; the only requirement is that individual variables be identified at declaration time with appropriate SI (or similar) units.

Keywords: compiler, plug-in, verification