

On the Correctness of Template Metaprograms*

Ádám Sipos^a, István Zólyomi^a, Zoltán Porkoláb^a

Dept. of Programming Languages and Compilers,
Fac. of Informatics, Eötvös Loránd University, Budapest
e-mail:{shp,scamel,gsd}@elte.hu

Abstract

C++ template metaprogramming (TMP) is a recently emerged programming paradigm that assists the creation of efficient code and flexible libraries. On the other hand, TMP is not yet widely used, due to the lack of coding standards and methodologies applicable when writing metaprograms.

In this paper we present methods for writing efficient and reliable metaprograms. We define the correctness of metaprograms and the possible types of failures to meet the specification. We describe the methods for creating metaprograms meeting their specifications. We analyze testing and bugfixing methods for them.

Unfortunately, even with the most modern programming methods bugs are not avoidable. Thus we describe the debugging methods applicable to metaprograms. As runtime debugging tools are not usable with metaprograms, we describe two new compile-time debugging methods. Our first method involves modification of the *g++* compiler to provide a compile-time trace about instantiated types. An other debugging method using a special language construct is also presented.

Keywords: C++, metaprogram, correctness, debugging

MSC: 68N19 Other programming techniques

Ádám Sipos

Pázmány Péter sétány 1/c., H-1117 Budapest, Hungary

István Zólyomi

Pázmány Péter sétány 1/c., H-1117 Budapest, Hungary

Zoltán Porkoláb

Pázmány Péter sétány 1/c., H-1117 Budapest, Hungary

*Supported by the Hungarian Ministry of Education under Grant FKFP0018/2002