A Preliminary Proposal for a Static if

Document #:	WG21/N3322 = PL22.16/12-0012
Date:	2011-12-11
Revises:	None
Project:	Programming Language C++

Reply to:	Walter E. Brown <wb@fnal.gov> FPE Dept., Scientific Computing Division Fermi National Accelerator Laboratory Batavia, IL 60510-0500

Contents

1 Introduction 1
2 Feature description 1
3 Prior art 2
4 An example 3
5 A second example 4
6 Acknowledgments 5

1 Introduction

This paper proposes a generalized compile-time conditional facility for possible future C++ standardization. In the remainder of this document, we refer to the proposed feature via a notional keyword static_if and refrain from any (bicycle-shed!) discussion of possible alternate nomenclature and keywords. [1]

2 Feature description

We envision high-level syntax and semantics for the proposed static_if analogous to those of the conventional if. Syntactically, there must be a predicate and two bodies, the second of which is taken to be empty if not explicitly provided:

```
static_if( predicate ) {
   body 1
}
else {
   body 2
}
```

Semantically, the predicate is evaluated, followed by a selection of one of the bodies according to the predicate's truth value. Our proposal differs from the conventional if in that all of this is required to happen during compilation rather than during execution.

> To ensure that a static_if's predicate can always be evaluated at compile-time, we will require that the predicate be a constant expression that can be converted to bool.

[1] For the record, the following alternatives have already been proposed by reviewers of preliminary drafts of this document: compile_if, only_if, enable_if / disable_if, if_, and (our current preference) if<...>.