A C++ ABI Test Suite

because

ABI bugs are a NIGHTMARE

Why should you test the ABI?
• To ensure release to release compatibility.
• To ensure compatibility with third party libraries.
• To ensure compatibility with tools that expect a specific ABI.

ABI bugs are a nightmare as they can hit you where you least expect and debuggers are often useless against them.

What does the ABI Test Suite do?
It tests a compiler’s implementation against the Itanium C++ ABI specification, by having C++ code that exercises various parts of the ABI specification, and compares the layout generated by the compiler-under-test to the ‘correct value’.

What does it test?
• Size and alignments of classes
• Offsets of fields and base classes
• Bit fields
• vtbl and VTT contents
• ctor and dtor vtables
• Name mangling
• Empty classes
• Thunks
• Init guard variables
• RTTI/typeinfo vars

Supporting x86 ILP32 and LP64
• All Itanium C++ ABI compilers; clang, gcc and others
• Both native and cross compilers
• Supports quirky compiler configurations
• Uses the lit framework
• Most tests were automatically generated
• Clang TOT passes

Getting started:
Put lit in PYTHONPATH and put FileCheck and clang in PATH
$ svn co http://llvm.org/svn/llvm-project/test-suite/trunk/ABI-Testsuite
$ python sample.py test
See README.txt, FAQ.pdf and the Design document for full details.