Distributed in the UK by

Phaedrus Systems

www.phaedsys.com

**SOLID SANDS**

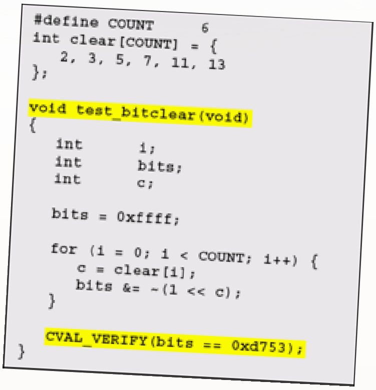
from Amsterdam is the one-stop shop
for C and C++ compiler testing,
validation and safety services.

Postbus 7897 | 1008 AB AMSTERDAM | The Netherlands

## THE IMPORTANCE OF COMPILER QUALITY

Compilers are essential during all stages of application development and systems integration. However, they are also very complex and a single error can create huge problems, both during development and during deployment of the generated code. Therefore it is extremely important to have confidence in the quality of the compiler. The potential cost of incorrectly generated code far outweighs the investment into a compiler quality control system.
Besides that, safety standards require that confidence in software tool-chains is brought to appropriate levels. Creating this confidence in a compiler starts with accurately testing the compiler for conformance, correctness and robustness.

## HOW YOU CAN ACHIEVE A HIGH LEVEL OF COMPILER QUALITY



```
#define COUNT      6
int clear[COUNT] = {
    2, 3, 5, 7, 11, 13
};

void test_bitclear(void)
{
    int     i;
    int     bits;
    int     c;

    bits = 0xffff;

    for (i = 0; i < COUNT; i++) {
        c = clear[i];
        bits &= ~(1 << c);
    }

    CVAL_VERIFY(bits == 0xd753);
}
```

The SuperTest compiler test and validation suite has proven to offer its users confidence and quality in the compilers they develop or use. Professional compiler developers, software quality engineers and compiler users all appreciate the many hand-crafted and generated test files in SuperTest, providing millions of conformance tests. This also includes many tests dealing with compiler internals such as analyses, transformations and optimizations. This set of tests and its special features are constantly growing. The framework includes both conformance and diagnostic tests. Diagnostic tests verify that the compiler reports errors for incorrect programs. The framework also has facilities to support selective testing of subsets of the suite, which can be used for example to re-run previously failed tests. The straightforward POSIX based user interface seamlessly integrates into any compiler development environment, allows for easy addition of new tests and provides cross platform validation support.

## UNIQUE FEATURES OF SUPERTEST

Next to what is expected from a compiler test and validation suite, like C language conformance, correctness and quality checks, SuperTest offers:

- Tests for C++ and C++11
- Remote and Parallel testing
- Clear reporting in HTML
- Easy addition of tests
- Powerful and flexible test generator
- .....

**ABI-Tester**
The ABI-Tester for C aims to expose errors in calling conventions and binary interfaces by generating pairs of files that contain the caller and callee respectively. It can be used to verify ABI-compliance within a single compiler, but also between different versions of the same compiler, or even different compilers for the same target.

**Depth suites**
In the C standard, the data model is left implementation defined. This means that test-suites have to limit their assumptions about arithmetic. SuperTest's Depth-suites are generated for specific data models and so have detailed knowledge about the boundaries of arithmetic. Depth-suites do extremely thorough and exhaustive testing of arithmetic operations with up to 5 operands. There are already about 30 different Depth-suites included in SuperTest, but on request we  can generate a customized suite.

**Tempest (TEMPlate Expander for SuperTest)**
This unique and flexible production-rule driven test generator can be used to generate random variations of tests. It allows the generation of complex programs that are predictable in their behavior due to the semantics of the generator-script-based productions rules. This enables specific compiler issues to be explored in more depth by generating a range of tests to extend the associated test-breadth, ensuring that there are no further related issues.

## SUPERTEST EXPERIENCE

“...Configuring SuperTest to do what we wanted was extremely straightforward, which meant that we were up and running with compiler testing within a few days.”

“...Right from the start we believed that the test coverage of SuperTest was significantly better than other compiler validation suites on the market, and certainly better the the GCC test suite. Over the past few years that has proved true, because SuperTest has identified several generic bugs in new releases of GCC that were not picked up by the GCC suite, some of which were relatively severe code generation issues.”

“...SuperTest helped us to find the problems in our compiler far more quickly than otherwise would have been possible. In comparison with the license and maintenance fees, we saved more than twice the amount in our efforts in improving the compiler quality.”

“...Although it’s not SuperTest’s primary function, we also use it to measure code size and performance. The beauty of using it for this purpose is that all of SuperTest’s validation tests are relatively small en terms of program size, which means that discovering why a section of code has changed either in size or performance is relatively straightforward.”

“...The test cases are well documented and tell you exactly where the problem is.

“...SuperTest is a really valuable tool for us. The verbose mode is wonderful for pointing me at the failing section of code. SuperTest found us implementation holes, implementation bugs, documentation errors, simulation bugs and many regressions. It is my primary regression test bench. I’d start using it very early in future compiler projects.”

“...SuperTest detected a lot of errors during our developments. Running the test suite successfully gave us enough confidence about the quality of each compiler version to be released to a next stage.”

“...The other big advantage in terms of timing was SuperTest’s fast run-time, which allowed us to achieve over 99% test coverage within a few weeks. That meant it offered very good value for money.”

“.. One of the strengths of SuperTest was that it delivered on its promise of being an ‘out-of-the-box’ solution.”