PARASOFT

Adopting GoogleTest for Safety-Critical Systems

GoogleTest—also known as gtest— is a robust, open source, xUnit-style C++ testing framework developed by Google and now carried out by the open-source community. As the de facto standard for C++ unit testing, it combines flexibility, portability, and seamless integration with modern toolchains.

However, in regulated industries, such as automotive, medical, aerospace, and rail, teams often need additional capabilities like compliance reporting, traceability, and code coverage to meet strict certification requirements.

Combined Solution Benefits

- **Unified testing workflow.** Run GoogleTest alongside C/C++test CT's unified testing capabilities in a single environment.
- **Enhanced coverage metrics.** C/C++test CT provides detailed coverage reports beyond GoogleTest's built-in capabilities.
- **Freedom from proprietary constraints.** Unlike some commercial frameworks that use abstracted, data-driven APIs, GoogleTest and C/C++test CT interact directly with source code, ensuring full compatibility with modern coding practices.
- **Scalability and team efficiency.** Readable and maintainable test code improves collaboration across large development teams, particularly in environments that demand continuous updates and shared ownership of tests.
- **Build system compatibility.** C/C++test CT and GoogleTest integrate seamlessly with distributed and modern build systems like Bazel and CMake, which are standard in many safety-critical embedded platforms.
- **Native C++ API design.** Clean, expressive syntax tailored for modern C++ makes it easier to write and maintain tests that involve STL containers, lambdas, and complex object hierarchies.

GoogleTest Key Features

Simplicity & Readability

- Supports test suites (TEST()) and fixtures (TEST_F()) for reusable setups.
- Writes tests in a clean, expressive syntax.

```
TEST(MathTest, Addition) {
  EXPECT_EQ(2 + 2, 4);    // Non-fatal assertion
}
TEST_F(DatabaseTest, Connection) {
  ASSERT_TRUE(db.connect());  // Fatal assertion
}
```

Cross-Platform Flexibility

- Supports unit tests, integration tests, and system tests.
- Works on Linux, Windows, macOS.
- Works with compilers like GCC, Clang, and MSVC.
- Works seamlessly with CMake, Bazel, and Makefiles.

Strong Assertions & Failure Messages

- Provides detailed error messages (e.g., Expected: 5, Actual: 3).
- Both non-fatal (EXPECT_*) and fatal (ASSERT_*) checks.
- Verifies if a program crashes as expected (e.g., ASSERT_DEATH()).
- Runs the same test logic with multiple inputs (TEST_P()).

Mocking Support

- Integrates with GoogleMock to simulate dependencies (e.g., APIs, objects)

Tool Integrations

- Works with IDEs (CLion, VS Code, Visual Studio).
- CI/CD friendly (Jenkins, GitHub Actions).
- Generates XML/JSON reports.

Active Community & Documentation

- Extensive docs
- Stack Overflow support

C/C++test CT Extends GoogleTest

Parasoft C/C++test CT (Continuous Testing) comes as a lightweight, command-line-driven tool designed for CI/CD pipelines and headless environments. The table below shows how it extends GoogleTest's functionality.

Capability	Added by C/C++test CT
Code coverage	Statement, branch, and MC/DC
Requirements traceability	Tag tests with IDs, for example, REQ-123, and generate reports
Compliance reporting	Automate reports for ISO 26262, code coverage, and so on
Seamless CI/CD integration	Jenkins, GitHub Actions, GitLab, Bazel, CMake

Our Expertise With GoogleTest

Working closely with leading automotive and aerospace organizations to successfully deploy GoogleTest in regulated environments, Parasoft's experience includes the following highlights.

- **Framework qualification leadership.** Developed a formal qualification process for GoogleTest, including over 430+ operational requirements and corresponding validation test cases.
- **Toolchain integration guidance.** Assisted clients in building complete toolchains around GoogleTest by integrating complementary tools for structural coverage, fault injection, and traceability reporting—tailored to meet ISO 26262, IEC 61508, and similar standards.
- **Tool safety manual.** Documented instructions on how to deploy the tool safely. The Tool Safety Manual also includes an analysis of all known and unresolved bugs, along with workarounds and mitigation strategies.

Key Considerations Before Adoption

Before adopting GoogleTest in safety-critical projects, teams should evaluate the following.

- **Qualification responsibility.** GoogleTest is not pre-qualified. However, in January 2026, Parasoft will be offering a TÜV SÜD certificate to organizations needing to satisfy ISO 26262, IEC 62304, IEC 61508, and EN 50716. C/C++test CT is TÜV SÜD certified.
- **Coverage, fault injection, and traceability gaps.** Addressing these requires using additional tools or frameworks, open-source or commercial, alongside GoogleTest.
- **Tool safety manual (TSM) and ongoing maintenance.** Safe usage requires a thorough TSM. You must maintain and update it with each new GoogleTest version and monitor for reported issues.
- **Requalification effort.** Adoption of new GoogleTest releases may require partial or full requalification, particularly if API or behavior changes impact existing requirements or validation results. Parasoft will perform this effort continuously for every C/C++test CT release.
- **Resource investment.** If you choose to qualify GoogleTest in-house, you'll need a skilled team dedicated to the certification process. In addition, relying on a trusted vendor can reduce risk, save time, and ensure a smoother path to compliance.

Get Started With Confidence

Contact us to discuss your compliance needs. Whether you're building embedded automotive systems or certified medical devices, ensure compliance and code quality in your GoogleTest-based testing workflow with Parasoft.

Parasoft Corporation
www.parasoft.com

101 E Huntington Drive
Monrovia, CA 91016 USA

Sales: 1-888-305-0041
International Sales: +1-626-256-3680