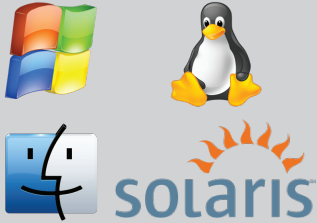


Understand

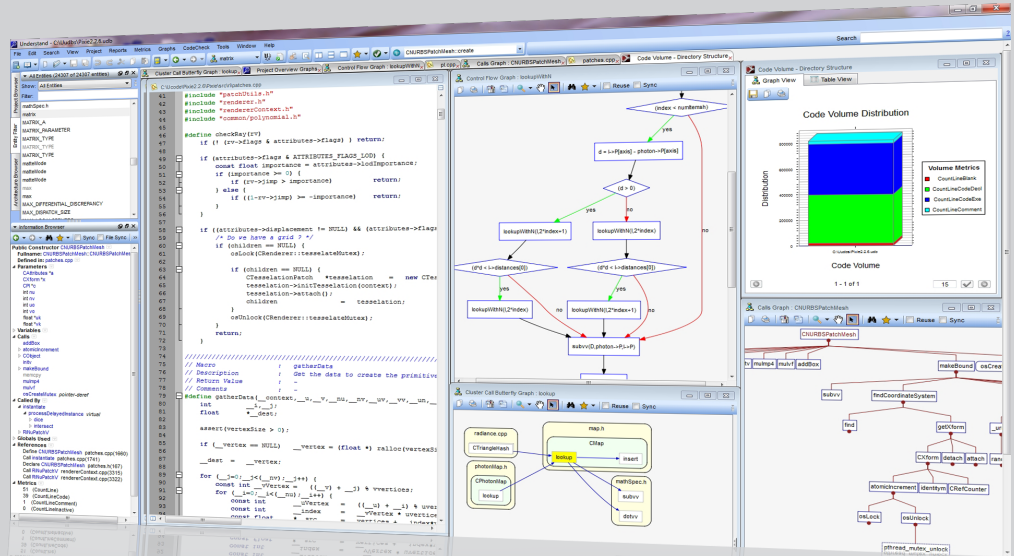
Source Code Analysis & Metrics

Testimonials



Supported Languages

C, C++, C#, Objective C, Ada,
Java, Pascal, Cobol, JOVIAL, VHDL,
FORTRAN, PL/M, Python, PHP,
HTML, CSS, JavaScript, XML



How much software are you maintaining?

Understand helps thousands of companies maintain impossibly large or complex amounts of source code. It parses source code for reverse engineering, automatic documentation, and calculating code metrics. We have versions for C, C++, C#, Objective C, Ada, Java, Pascal, Cobol, JOVIAL, VHDL, FORTRAN, PL/M, Python, PHP, HTML, CSS, Javascripts, XML. Multi-million SLOC projects are common with our users.

Probably more than you think...

You may not know it, but software maintenance is actually your biggest software development cost. Maintenance begins, well... right after the code is created. Even before you ship your product, you are maintaining the code.

Is it big? Is it old? Somebody else write it?

If you answered yes to any of these questions, **Understand** will help you immediately and a lot. Download and try it, it costs nothing, and you can see for yourself if the powerful user testimonial on the right, will apply to you as well.

Indispensable...

"I have found **Understand** to be an indispensable tool for understanding and enhancing legacy code comprising over a million lines of code. This is especially true for code dealing with infrastructure where changes can have significant rippling affects within the system. I have personally used **Understand** on consumer based financial software such as *Quicken* and *QuickBooks* where quality and reliability is a must. In fact, significant online banking changes to *Quicken 2007* and *QuickBooks 2008* would not have been done if the team didn't have a tool such as **Understand**."

www.scitools.com

Enforcing Coding Standards...

I'm working on a project which established a list of coding standards. The project leaders then purchased a tool that they had contemplated on using to validate the code to the coding standards. When the "tool" was installed & configured it missed out on reporting a majority of this projects coding 'standards'. I found that using an **Understand** Perl script for C++ that reports the entities properties and importing the results file to a database, I could run a unique query to identify any non-conforming entity (for example all pointers need to start with "p_"). For every problem there is a solution and Understand turned out to be **THE** solution...

--Avionics Engineer
The Boeing Company

Huge Time Saver...

I've often had to switch projects between completely different software platforms. Normally it takes a day or two to get up to speed on a particular SW Architecture, but with **Understand** this is shortened to an hour or less.

I've also seen how long it takes to find a function or variable in several other editors. **Understand** is an order of magnitude faster.

As far as I am concerned, **Understand** is THE tool for development in C.

--Avionics Engineer
BAE Canada

IVV Use...

My team and I have been using **Understand** for performing Independent Validation and Verification (IV & V) on hundreds of thousands of lines of Ada, C++, and FORTRAN code.

Understand is an essential tool for what we do, it helps us identify coding problems and inefficiencies that can be reported to the software developers, producing a better end product.

Another great benefit is the excellent user support provided by **SciTools**.

--USAF Software Engineer

Very Useful...

This software saved my life! I was assigned to fix a bad coding job where the programmer used 4 letter acronyms for variables and functions. Nuts. Now it helps me debug code without compiling and navigate with ease. Accessing Clearcase through Explorer is SLOW. But, if you setup scripts using **Understands** scripts, you can checkout, checkin, diff, tree in no time! Just wish they had a tool for Verilog!

--Embedded Software Engineer

Fantastic Tool...

Understand for Ada is a fantastic tool. It produces good quality static analysis reports, a must have for the company dealing with real time product/services departments.

--Software Engineer, Aersospace industry

Excellent Support...

Recently, we were trying to use the **Understand** for Ada program and the "dead code" perl script to determine exactly which Ada files we had running in each of our processors, but we were missing instantiations of generics. We wrote to STI support services and within a day or two, we had a modified perl script that produced the results we were seeking.

Thanks **STI** for the kind of support every company should provide.

--Software Engineer, Raytheon

Fastest way to make sense of existing code...

We have been using **Understand** for several years at my company, and it has been spreading through the company (via a floating 5-user license). It is by far the fastest way I have ever found to try to make sense of an existing code base, which is something I spend a lot of time doing. I recommend this product with great enthusiasm!

I also will add that the tech support I have gotten, even when I have let my maintenance lapse, is above reproach. Fast and direct.

--Software Engineer, General Dynamics