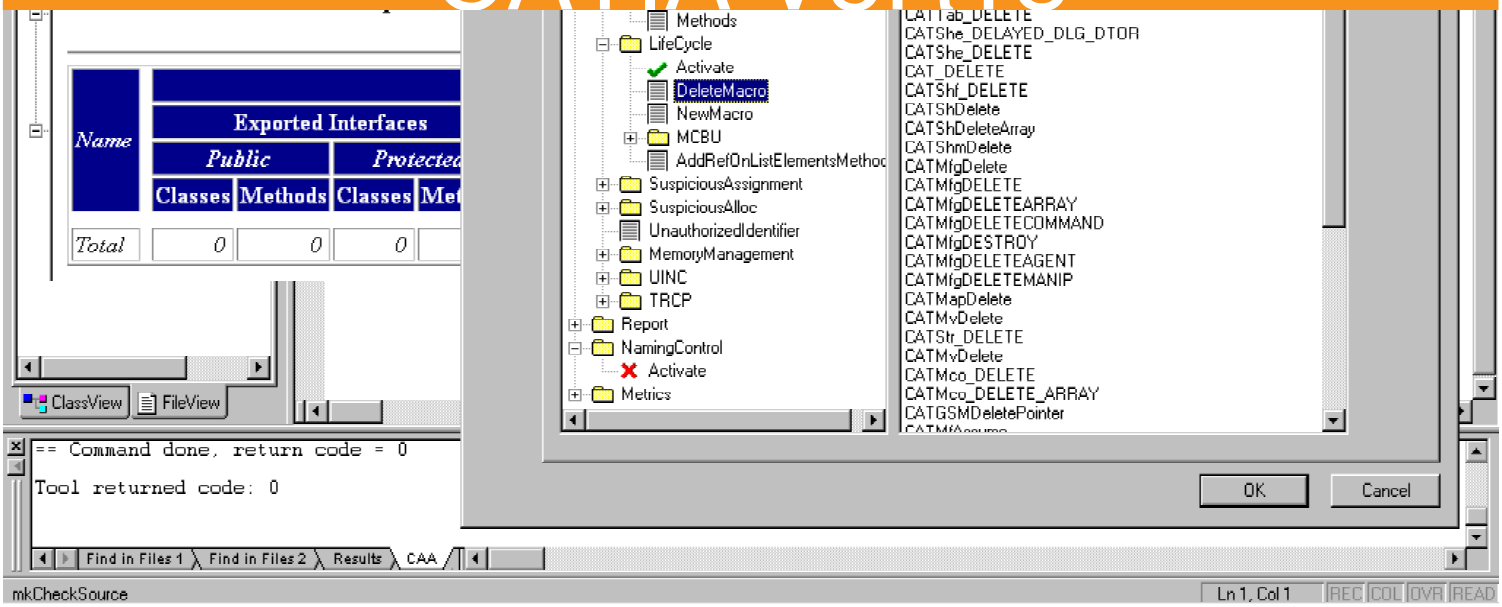
CAA-RADE

CAA C++ SOURCE CHECKER (CSC)

CATIA V5R18



CAA-RADE

CAA - C++ Source Checker

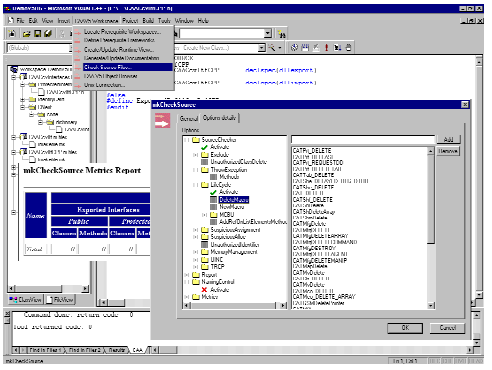
CAA - C++ Source Checker (CSC) facilitates test and quality control tasks critical to the efficient development of quality software. It operates from the source code application.

Product overview

CAA - C++ Source Checker (CSC) contributes to enhance the global quality objective of CAA V5 applications. Operating at the source stage in the application development cycle, early checking against C++ coding rules will ensure better stability and reduce defect. As for the other RADE products, it operates on both Windows and UNIX platforms with the same characteristics.

Product Highlights

- Same characteristics under both Windows and UNIX
- Automatic check of C++ V5 coding rules
- Memory leaks debugging for object modeler
- Provides C++ source parser
- Full HTML report with hyperlink to faulty C++ source
- Extension of C++ coding rules related to null pointer checks:
- Memory management rules check improvement
- Enables recognition of macros that defined memory release
- Windows XP support for build time
- This enable the usage of CAA - C++ SOURCE CHECKER (CSC) on Windows operating system
- Filters to disable reporting of specific detected errors
- Extend parser capabilities to support Macro expansion for more than one level
- Detection of brackets after condition



- Report of header files that should be included directly in a cpp file and that are included indirectly

Product Key Customers Benefits

Automatic check of C++ V5 coding rules

Different C++ rules are checked against the sources code. Thanks to these rules, time spent to debug applications is drastically reduced and the quality of the code is improved. Developers are provided with a number of rules that help them reduce memory related bugs in domain such as application memory management, call back mechanism usage, exception handling and C++ programming rules. These sensitive checks ensure a better control of application quality and globally decrease the number of bugs related to memory corruption. Another aspect when checking sources is the ability to analyze discrepancies in the usage of C++ null pointers. This rule check ensures a better control on the number of core dumps occurring during the execution of the

application

Memory leaks debugging for Object Modeler

Another set of C++ rules are provided that permit easy and fast detection of memory leaks within the tested code. As debugging memory leaks can be very time consuming, this automatic detection addresses a major need for large application project as well as small application development.

Provides C++ source parser

It ensures that sources are well checked against the provided rules. The parser acts as a front task compiler, it generates the syntax tree of the source, creates the table of symbols. Multiple sources and frameworks checking can be handled across different workspaces, thus taking into account potential external prerequisite frameworks. Parsing of C++ code can operate from the workspace to be analyzed and a provided pattern-matching list of frameworks inside this workspace. Developers can also start the parsing providing only a pattern-matching list of source to be analyzed. Recursive macro expansion are also supported by the parser.

Full HTML report with hyperlink to faulty C++ source

Provides analysis HTML report ready to be exploited. Includes full features hyperlinks, which allows a deep analysis from framework to faulty C++ source line. Error detection is highlighted at all level from framework to module to C++ source line and word. This ensures an easy and reliable analysis of the checked code that allows a fast correction.

ABOUT CATIA V5R18

CATIA is Dassault Systemes' PLM solution for digital product definition and simulation.

plm.3ds.com/CATIA