

Please note that, following the withdrawal of Carbide.vs, Forum Nokia no longer supports the Microsoft Visual Studio IDE for Symbian C++ development. For more information see the [Carbide.vs withdrawal page](#) on Forum Nokia.

Contents

- [1 Introduction](#)
- [2 Installation](#)
- [3 Visual Studio.NET Intergration](#)
- [4 Executing](#)
- [5 Also see](#)

Introduction

SymScan is an official and very useful tool from Symbian C++ Developers. It's the improved successor of [LeaveScan](#) and can scan your own source code files, to find out possible problems related to using the correct naming conventions and the cleanup stack. In detail, it helps against the following:

- **Deprecated literal macro (?_L(?)):** Instead, you should use the `_LIT()`-macro, as it's more efficient.
- **Correct usage of the Cleanup:** stackAre objects that you create for automatic variables correctly put on the cleanup-stack as well as removed and destroyed again?
- **Opening and Closing of 'R' classes:** R-Classes are connections to servers (file server, ...)
- **Highlighting functions marked as non-leaving which can leave:** In case you forgot to add the `?L?`-prefix to a function name of a method that can cause a leave.
- **Correct use of descriptors in function call:** Is the usage of descriptors as parameters for functions correctly executed?

Installation

Download the tool from the URL below and install it. It will automatically place itself in the following directory: `C:\Program Files\Common Files\Symbian\tools?`

- [SymScan Binary](#)
- [SymScan Source](#)

Visual Studio.NET Intergration

To be able to check a file directly from VS.net, you can create an individual menu command. This can be done as follows:

Tools -> External Tools -> Add

Add the following information:

- Title: SymScan
- Command: C:\Program Files\Common Files\Symbian\tools\SymScan.exe
- Arguments: \$(ItemPath)
- Activate **Use Output Window**

Executing

When your own .cpp file is open and active, execute the tool through Tools -> SymScan. The program will check the currently open file and write the results to the output window.

Also see

- [Checking your code with SymScan in Carbide.c++](#)
-