# Sulprobil.xll Installation Guide

This guide is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See my disclaimer at http://sulprobil.com/html/disclaimer.html, too, please.

## 1. Introduction

This is an installation guide to install the Sulprobil Excel© add-in development environment.  You will only need this if you are interested in recreating the Sulprobil add-in independently from scratch or if you need to port or to enhance it for your individual needs.  It is my personal opinion that I should only publish executable programs if I also provide a proper license (in case I sell it) or the underlying source code together with the know-how to create the executable from the sources.

So if you just need the Sulprobil add-in as a ready-to-use executable, go to my website www.sulprobil.com, copy the file Sulprobil.xll into your local add-in directory, reload Excel©, press ALT + T, then ALT + I and tick the Sulprobil box. Then you can use all add-in functions I provided.

Why programming an Excel add-in?  During decades of using Excel I developed and I collected a nice heap of small and helpful Excel VBA programs.  They can be easily copied and combined in any application I develop.  But if I want to use the most recent version of any code easily I need it at one specified location and in one container.

Why using the Microsoft Visual Studio C++ 2008 Express compiler?  I do not like the idea of an Excel VBA add-in.  It can be changed too easily and you cannot see immediately if somebody did.  Microsoft offers the C++ 2008 Express compiler for free.  It can be installed easily, so I thought it's a good start.  I might switch to the GNU C++ compiler at a future stage, though.  This would not be a decision about runtime speed.  I liked the old UNIX programming environment:  There is a nice tool called Aegis which can enforce quality by providing a good version control and configuration management.  Unfortunately it's not really available in the Microsoft Windows environment.  Maybe I will even develop Excel add-in functions in the UNIX programming environment.  Let's wait and see.

Why using XLW?  XLW is a free wrapper for the Excel application programming interface (API).  I do not need to fumble around with complex Excel interfaces.  I just use the simple XLW interfaces and I can focus on my add-in functionality.  XLW is available at http://xlw.sourceforge.net.

Why using Boost libraries?  The C++ Boost libraries offer a very good functionality in addition to the C++ STL.  They are of good quality because they are peer-reviewed.  Boost has become sort of a semi-standard.  You can get it for free at www.boost.org but I got it from http://www.boostpro.com/download because I thought that's easier to start with (you need to register but it was free as well when I did it).

# Sulprobil.xll Installation Guide

This guide is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See my disclaimer at http://sulprobil.com/html/disclaimer.html, too, please.

## 2. Installation & Configuration

a)  Install Microsoft Visual C++ 2008 Express edition: www.microsoft.com/express/download

b)  Install Boost libraries, version 1.40: www.boostpro.com/download  (the official site is www.boost.org but I found it easier to use boostpro)

c)  [Optional, please see later step to define additional include directories] In Visual C++ 2008 select Tools / Options… / Projects and Solutions / VC++ Directories / Show Directories for Include Files and insert new line for your Boost include folders, for example "C:\Program Files\boost\boost_1_40"

d)  Install XLW, version 4.0.0f0: http://xlw.sourceforge.net

e)  Extract XLW xll template: Start / All programs / XLW / xlw-4.0.0f0 / xlw / Extract XLW xll template

f)  Uncomment the line
    #define USE_XLW_WITH_BOOST_UBLAS
    in "…\XLW\xlw-4.0.0f0\xlw\include\xlw\MyContainers.h"
    for example "C:\Program Files\XLW\xlw-4.0.0f0\xlw\include\xlw\MyContainers.h"
    or "E:\Programme\XLW\xlw-4.0.0f0\xlw\include\xlw\MyContainers.h"

g)  [Please notice that all remaining steps are also to be repeated In case of an updated Sulprobil version]

h)  Unzip Sulprobil C++ source code from www.sulprobil.com into XLL_Project directory, for example:
    [Vista: C:\Users\<user name>\Documents\XLL_Project]
    [XP: E:\Dokumente und Einstellungen\<user name>\Eigene Dateien\XLL_Project]

i)  [Optional, please see previous step to define directories for include files] In Visual C++ 2008 ensure that folder Source files in Solution Explorer is selected, then go to Project / XLL Properties / Configuration Properties / C/C++ / General / Additional Include Directories and make sure it's showing "E:\Programme\boost\boost_1_40;"$(XLW)\xlw\include";.\source" [Instead of E:\Programme it should be the correct path on your computer…]

j)  In Visual C++ 2008 ensure that folder Source files in Solution Explorer is selected, then go to Project / XLL Properties / Configuration Properties / Linker / General / Additional Library Directories and make sure it's showing "$(XLW)\xlw\lib";"C:\Program Files\boost\boost_1_40\lib" [Instead of C:\Program Files it should be the correct path on your computer…]

k)  Double-click on file Sulprobil.sln in your XLL_Project directory. Visual C++ 2008 will start with the Sulprobil project loaded

l)  In Visual C++ 2008 press F7 (Build / Build solution). The successful compilation log should look like

```
1>------ Build started: Project: RunInterfaceGenerator, Configuration: Release Win32 ------
1>Performing Makefile project actions
1>Microsoft (R) Program Maintenance Utility Version 9.00.30729.01
1>Copyright (C) Microsoft Corporation.  All rights reserved.
1> ""C:\Program Files\XLW\xlw-4.0.0f0\xlw\build\vc9\Release\InterfaceGenerator.exe""
source\cppinterface.h source\xlwWrapper.cpp
1>file has been read in
1>file has been tokenized
1>file has been stripped
1>file has been function modeled
1>file has been function described
1>new file is a vector
1>all done
1> ECHO flagged > Release\buildflag
1>Build log was saved at "file://c:\Users\Bernd\Documents\XLL_Project\Release\BuildLog.htm"
1>RunInterfaceGenerator - 0 error(s), 0 warning(s)
2>------ Build started: Project: XLL, Configuration: Release Win32 ------
2>Compiling...
2>xlwWrapper.cpp
```

# Sulprobil.xll Installation Guide

This guide is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See my disclaimer at http://sulprobil.com/html/disclaimer.html, too, please.

```
2>sbXtractNum.cpp
2>sbWeekNum.cpp
2>sbWAvg.cpp
2>sbVersion.cpp
2>sbVector.cpp
2>sbSplit.cpp
2>sbReverse.cpp
2>sbNSig.cpp
2>sbNRN.cpp
2>sbMmult.cpp
2>sbMin.cpp
2>sbMax.cpp
2>sbCat.cpp
2>Generating Code...
2>Linking...
2>   Creating library .\Release/Sulprobil.lib and object .\Release/Sulprobil.exp
2>Embedding manifest...
2>Build log was saved at "file://c:\Users\Bernd\Documents\XLL_Project\Release\obj\BuildLog.htm"
2>XLL - 0 error(s), 0 warning(s)
========== Build: 2 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
```

m) Copy the Sulprobil add-in Sulprobil.xll into your local add-in directory:

[XP: Copy "E:\Dokumente und Einstellungen\<User name>\Eigene Dateien\XLL_Project\Release\Sulprobil.xll" into your Excel add-in directory „E:\Dokumente und Einstellungen\<User name>\Anwendungsdaten\Microsoft\AddIns"]

[Vista:  Copy "C:\Users\<user name>\Documents\XLL_Project\Release\Sulprobil.xll" into "C:\Users\<User name>\AppData\Roaming\Microsoft\AddIns"]

Please notice that your subdirectory "Application Data" (or "Anwendungsdaten") might be hidden. In this case select Option "Show all files and directories" in Windows explorer.