How to set-up an ProMalDes-development framework with VC (v12.4.2019)

1. Download Visual Studio Community

- https://visualstudio.microsoft.com/de/vs/community/
- Install also the English language package;

2. Download QT-version for MSVC (open source version; 5.12.1 msvc2017_64)

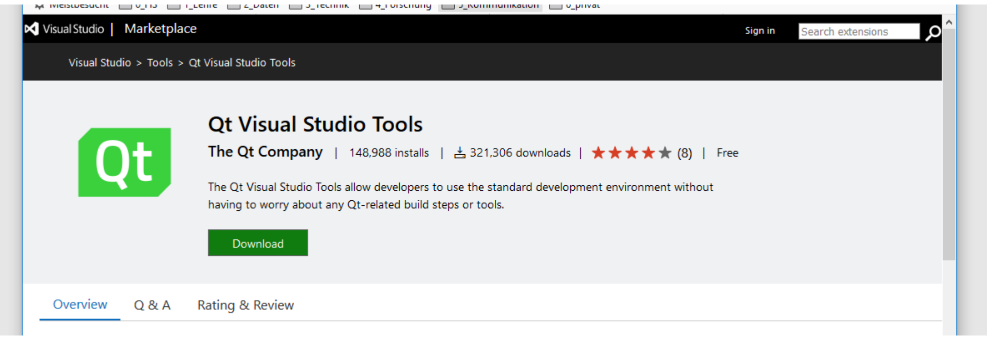
- https://www.qt.io/download

3. Download QWT (currently 6.1.0)

- https://sourceforge.net/projects/qwt/files/qwt/6.1.0/
- It must be compiled by VS; I can also deliver the binaries

4. Download QT-add in for VC and install it

- https://marketplace.visualstudio.com/items?itemName=TheQtCompany.QtVisualStudioTools-19123



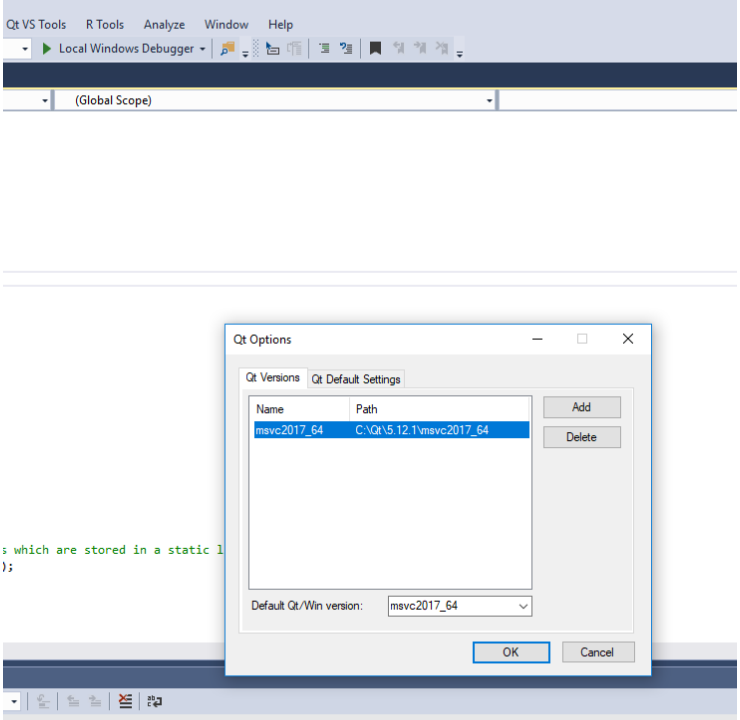
Visual Studio | Marketplace
Sign in
Search extensions
Visual Studio > Tools > Qt Visual Studio Tools
Overview
Q & A
Rating & Review
Download

Qt Visual Studio Tools

The Qt Company | 148,988 installs | 321,306 downloads | ★★★★★ (8) | Free

The Qt Visual Studio Tools allow developers to use the standard development environment without having to worry about any Qt-related build steps or tools.

- Set the available QT-versions in MSVC in the *QT VS Tools*



Qt Options

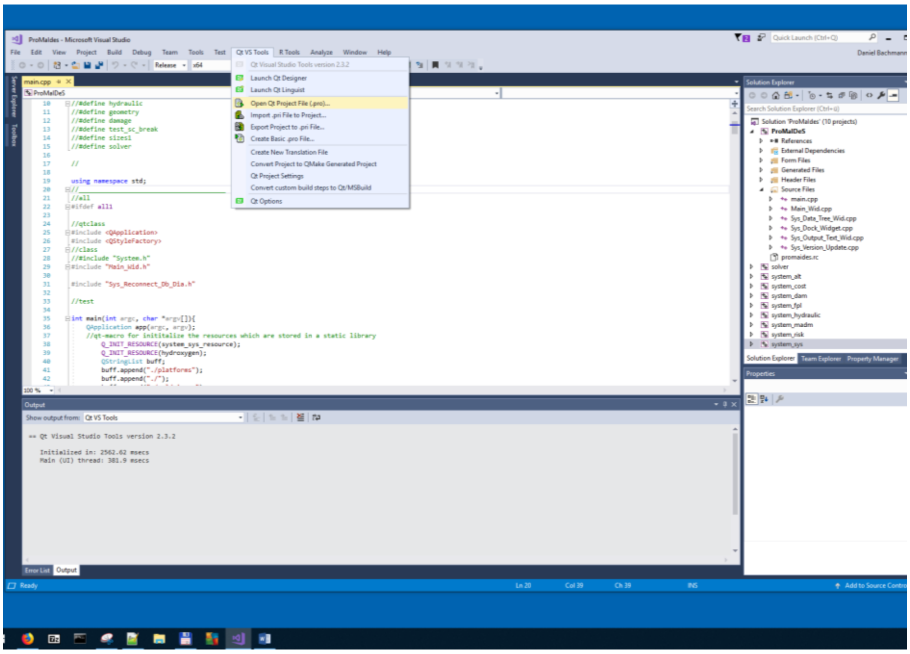
Qt Versions | Qt Default Settings

Name	Path
msvc2017_64	C:\Qt\5.12.1\msvc2017_64

Default Qt/Win version: msvc2017_64

```
s which are stored in a static l
);
```

5. Load step by step the pro-files of each module into VS

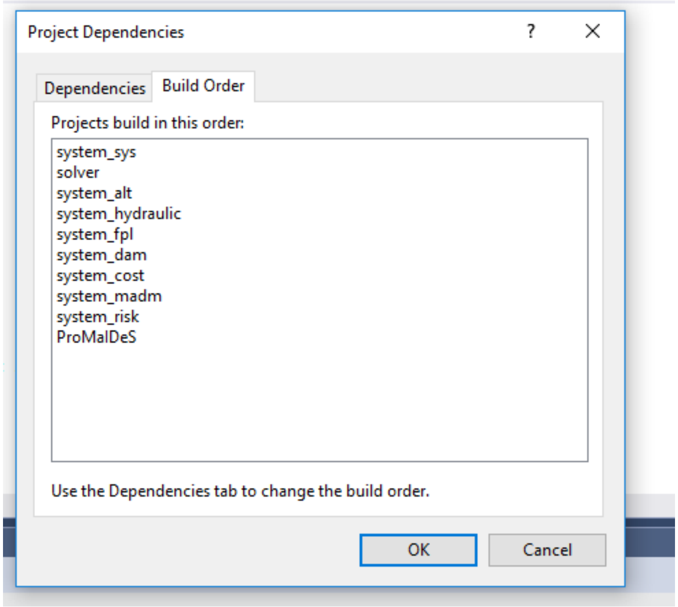


```
//#define hydraulic
//#define geometry
//#define damage
//#define test_sc_break
//#define sizes1
//#define solver

//

using namespace std;
//_______________________________________
//all
#ifdef all1

//qtclass
#include <QApplication>
#include <QStyleFactory>
//class
//#include "System.h"
#include "Main_Wid.h"

#include "Sys_Reconnect_Db_Dia.h"

//test

int main(int argc, char *argv[]){
    QApplication app(argc, argv);
    //qt-macro for inititalize the resources which are stored in a static library
        Q_INIT_RESOURCE(system_sys_resource);
        Q_INIT_RESOURCE(hydroxygen);
        QStringList buff;
        buff.append("./platforms");
        buff.append("./");
```

Show output from: Qt VS Tools

```
== Qt Visual Studio Tools version 2.3.2

   Initialized in: 2562.62 msecs
   Main (UI) thread: 381.9 msecs
```

- Start by system_sys (all project will produce static libs, just the ProMalDes.pro file will generate an executable)
- Use via *QT VS Tools/Open QT project file…*
- Set the project dependency; the required build order is shown below



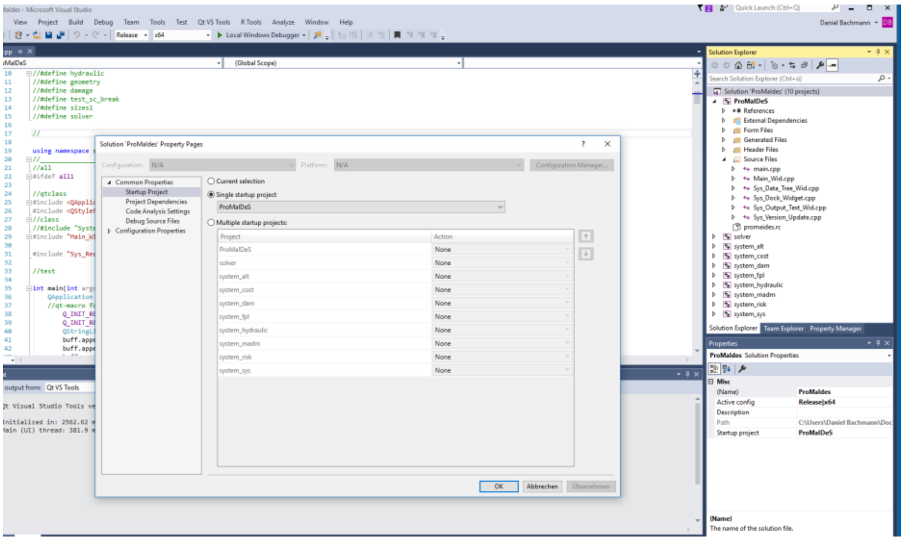
Projects build in this order:

system_sys
solver
system_alt
system_hydraulic
system_fpl
system_dam
system_cost
system_madm
system_risk
ProMalDeS

Use the Dependencies tab to change the build order.

- Try to compile each project step-by-step

- Set the ProMalDes-project as “Start-up project” (e.g “solution property pages”)



Solution 'ProMaldes' Property Pages

Configuration: N/A Platform: N/A

○ Current selection

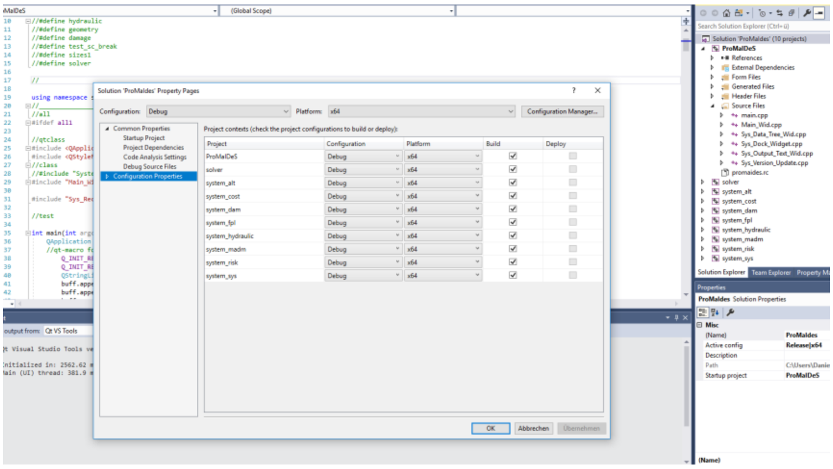
● Single startup project

ProMalDeS

○ Multiple startup projects:

Project	Action
ProMalDeS	None
solver	None
system_alt	None
system_cost	None
system_dam	None
system_fpl	None
system_hydraulic	None
system_madm	None
system_risk	None
system_sys	None

- Here you can specify further build option



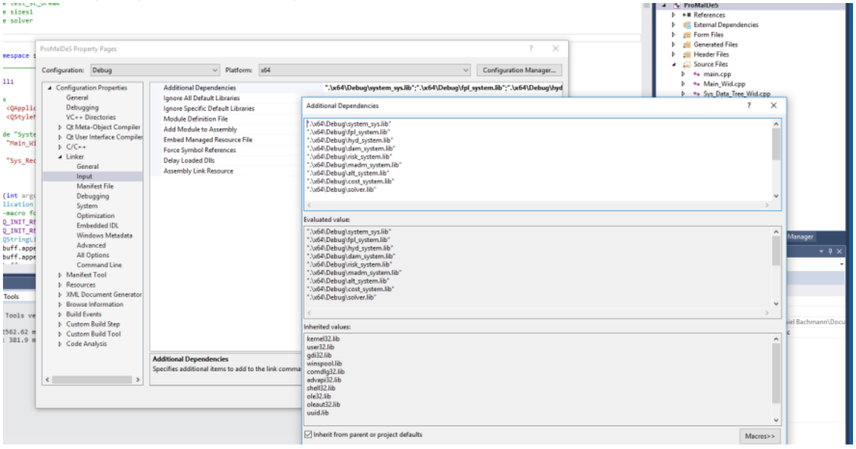
Solution 'ProMaldes' Property Pages

Configuration: Debug Platform: x64

Project contexts (check the project configurations to build or deploy):

Project	Configuration	Platform	Build	Deploy
ProMalDeS	Debug	x64	☑	□
solver	Debug	x64	☑	□
system_alt	Debug	x64	☑	□
system_cost	Debug	x64	☑	□
system_dam	Debug	x64	☑	□
system_fpl	Debug	x64	☑	□
system_hydraulic	Debug	x64	☑	□
system_madm	Debug	x64	☑	□
system_risk	Debug	x64	☑	□
system_sys	Debug	x64	☑	□

- Beware that after an import of the ProMaldes-project file you have to delete in the Linker-properties the qwt.lib



Additional Dependencies ".\x64\Debug\system_sys.lib";".\x64\Debug\fpl_system.lib";".\x64\Debug\hyd

Additional Dependencies

```
".\x64\Debug\system_sys.lib"
".\x64\Debug\fpl_system.lib"
".\x64\Debug\hyd_system.lib"
".\x64\Debug\dam_system.lib"
".\x64\Debug\risk_system.lib"
".\x64\Debug\madm_system.lib"
".\x64\Debug\alt_system.lib"
".\x64\Debug\cost_system.lib"
".\x64\Debug\solver.lib"
```

Evaluated value:

```
".\x64\Debug\system_sys.lib"
".\x64\Debug\fpl_system.lib"
".\x64\Debug\hyd_system.lib"
".\x64\Debug\dam_system.lib"
".\x64\Debug\risk_system.lib"
".\x64\Debug\madm_system.lib"
".\x64\Debug\alt_system.lib"
".\x64\Debug\cost_system.lib"
".\x64\Debug\solver.lib"
```

Inherited values:

```
kernel32.lib
user32.lib
gdi32.lib
winspool.lib
comdlg32.lib
advapi32.lib
shell32.lib
ole32.lib
oleaut32.lib
uuid.lib
```

☑ Inherit from parent or project defaults

Additional Dependencies
Specifies additional items to add to the link comma

- Project dependency is as follow:
  - ProMaideS: to all
  - Solver: to none
  - System_alt: system_dam, system_fpl, system_hydraulic
  - System_cost: system_alt

- system_dam: system_hydraulic, system_sys
- system_fpl: system_sys
- system_hydraulic: solver, system_sys
- system_madm: system_cost, system_risk
- system_risk: system_dam,system_fpl,system_hydaulic, system_sys
- system_sys: to none

6. Deploying:

for **deploying** the tool following files should by in folder (I'm not sure if all are required!, Should check it on a clean PC):

- imageformats\ *(this one comes from the Qt/plugin folder)*
- platforms\ *(this one comes from the Qt/plugin folder)*
- ProMaIDeS.tlog\
- sqldrivers\ *(this one comes from the Qt/plugin folder)*
- alt_system.lib
- cost_system.lib
- dam_system.lib
- fpl_system.lib
- hyd_system.lib
- icudt52.dll
- icuin52.dll
- icuuc52.dll
- libbz2-1.dll
- libeay32.dll
- libfreetype-6.dll
- libgcc_s_dw2-1.dll
- libgcc_s_seh-1.dll
- libglib-2.0-0.dll
- libgomp-1.dll
- libharfbuzz-0.dll
- libiconv-2.dll
- libintl-8.dll
- libjpeg-8.dll
- libmysql.dll
- libpcre16-0.dll
- libpng16-16.dll
- libpq.dll
- libstdc++-6.dll
- libwinpthread-1.dll
- madm_system.lib
- main.obj
- Main_Headers_Precompiled.h.obj
- Main_Wid.obj
- moc_Main_Wid.obj
- moc_Sys_Data_Tree_Wid.obj
- moc_Sys_Dock_Widget.obj
- moc_Sys_Output_Text_Wid.obj
- ProMaIDeS.log
- promaides.res
- ProMaIDes_vc.exe

- ProMaIDes_vc.pch
- ProMaIDes_vc.Build.CppClean.log
- Qt5Core.dll
- Qt5Gui.dll
- Qt5OpenGL.dll
- Qt5PrintSupport.dll
- Qt5Sql.dll
- Qt5Svg.dll
- Qt5Widgets.dll
- Qt5Xml.dll
- qt_work.log
- qwt.dll (*this one comes from the compiled qwt-directory*)
- risk_system.lib
- solver.lib
- ssleay32.dll
- Sys_Data_Tree_Wid.obj
- Sys_Dock_Widget.obj
- Sys_Output_Text_Wid.obj
- Sys_Version_Update.obj
- system_sys.lib
- zlib1.dll

the QT-libs should be also copied from Qt folder

7. Debugging

the same procedure you have to do for your debug folder; especially the sqldrivers, images and platform-folders are important!; be aware that you have to change the working directory of your Debug version in the Project-properties (Debugging) of the ProMaIDes-project to the folder, where the folders and libs are stored!

Good luck!