Subject: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Fri, 30 Apr 2021 12:02:06 GMT
View Forum Message <> Reply to Message

Importing a large library with "import.ext" I got a couple of probmems:

1) Some files need to be excluded / included depending on platform.
A nice way would be something like

```
files(PLATFORM_POSIX)
    files.cpp
    dir1/*.cpp
;
exclude(PLATFORM_WIN32)
    an_unix_file.cpp
;
```

2) A big problem is that object files are put ALL in ONE folder. If the library
has something like this:

```
    dir1/MyNiceModule.cpp
    dir2/MyNiceModule.cpp
```

build will fail because second MyNiceModule.o is placed in same output folder and overwrites first one.
The solution could be to put .o files in subfolder or add a prefix like dir1_MyNiceModule.o.

Ciao

Massimo

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Sat, 01 May 2021 16:19:57 GMT
View Forum Message <> Reply to Message

I am all for it (after 2021.1 release).

If you could look into it (ide/Builders/CppBuilder.cpp:262), I will be happy to integrate the patch. If not, I will do it myself, some nice sunny or cloudy day in the future when I have time and mood...

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"

Posted by mdelfede on Sun, 02 May 2021 16:40:41 GMT
View Forum Message <> Reply to Message

For point 1 I can try it... just a question : the directives in import.ext are processed sequentially ?
I mean :

files somefolder/*.cpp;
exclude somefolder/platforms/*;
files somefolder/platforms/posix/*.cpp;

will pull in all somefolder cpp files, excluding somefolder/plaforms folders but including somefolder/plaforms/posix folders ?

For point 2, I see it more difficult without tampering with all build system.

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Mon, 03 May 2021 06:45:01 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Sun, 02 May 2021 18:40For point 1 I can try it... just a question : the directives in import.ext are processed sequentially ?
I mean :

files somefolder/*.cpp;
exclude somefolder/platforms/*;
files somefolder/platforms/posix/*.cpp;

will pull in all somefolder cpp files, excluding somefolder/plaforms folders but including somefolder/plaforms/posix folders ?

Yes. I gave you a pointer to the code, did not I?

Quote:
For point 2, I see it more difficult without tampering with all build system.

Yes. Which is probably a good thing.

Mirek

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by Oblivion on Mon, 03 May 2021 08:15:49 GMT
View Forum Message <> Reply to Message

This is a good idea, but for the record, there is already a serious problem with unrecognized keywords/flags/text in ext files, which I had reported here.

Best regards,
Oblivion

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Mon, 03 May 2021 23:30:49 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 03 May 2021 08:45

Yes. I gave you a pointer to the code, did not I?

Yep, and the part 1 is almost done. I just need to know which flags are available at build time and how to read them... if you
can point me in the right direction you'll spare me some time.
By now I added an optional (flags) part, where 'flags' can be any expression, for example:

TRUE
true
1
FALSE
false
0
!flag && (flag1 || flag2)
!flag && flag1 || flag2

with ! (or ~) is the negation, && is AND, || is OR, precedence is UNARY > && > ||.

For example:

files mypath/*.cpp;

Behaves as usual.

files(flagPosix && !flagSomething) mypath/*.cpp;

fetches the files only if flagPosix is true and flagSomething is false.
This applies also to other items (include, exclude, etc).

Just need to evaluate the flag(s) and it's complete.

Ciao

Max

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Tue, 04 May 2021 05:50:28 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Tue, 04 May 2021 01:30mirek wrote on Mon, 03 May 2021 08:45

Yes. I gave you a pointer to the code, did not I?

Yep, and the part 1 is almost done. I just need to know which flags are available at build time and how to read them... if you
can point me in the right direction you'll spare me some time.
By now I added an optional (flags) part, where 'flags' can be any expression, for example:

TRUE
true
1
FALSE
false
0
!flag && (flag1 || flag2)
!flag && flag1 || flag2

with ! (or ~) is the negation, && is AND, || is OR, precedence is UNARY > && > ||.

For example:

files mypath/*.cpp;

Behaves as usual.

files(flagPosix && !flagSomething) mypath/*.cpp;

fetches the files only if flagPosix is true and flagSomething is false.
This applies also to other items (include, exclude, etc).

Just need to evaluate the flag(s) and it's complete.

Ciao

Max

Is it using MatchWhen? It should.

Flags will need to be passed as additional parameter.

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Tue, 04 May 2021 06:10:17 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 04 May 2021 07:50
Is it using MatchWhen? It should.

Nope, I wrote a quick-and-dirty expression evaluator on the fly... had no idea
about MatchWhen... I see that I reinvented the wheel, as usual.
Btw, my flags fetching question is still open... where are available flags ?

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Tue, 04 May 2021 06:18:32 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Tue, 04 May 2021 08:10mirek wrote on Tue, 04 May 2021 07:50
Is it using MatchWhen? It should.

Nope, I wrote a quick-and-dirty expression evaluator on the fly... had no idea
about MatchWhen... I see that I reinvented the wheel, as usual.
Btw, my flags fetching question is still open... where are available flags ?

Quote:
Flags will need to be passed as additional parameter.

Check ide/Builders/GccBuilder.cpp:88

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Tue, 04 May 2021 07:44:03 GMT

mirek wrote on Tue, 04 May 2021 07:50
Is it using MatchWhen? It should.

Mhhhh... You mean I shall use ReadWhen to extract the expression in string format and then use MatchWhen over it ?
My way used the parser directly.
What do you thing about exposing

```
static bool sMatchOr(CParser& p, const Vector<String>& flag)
```

As public function, for example overloading MatchWhen:

```
bool MatchWhen(CParser& p, const Vector<String>& flag)
{
    return sMatchOr(p, flag);
}
```

So it's possible to do :

```
bool CheckImportCondition(CParser &p, const Vector<String>& flag)
{
 // no condition == true
 if(!p.Char('('))
  return true;
 bool res = MatchWhen(p, flag);
 p.PassChar(')');
 return res;
}
```

??

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Tue, 04 May 2021 21:11:35 GMT

Ok... first mod is done.
With this patch ide accepts optional (flag) inside import.ext file.

I don't remember the way to contribute, so I'm attaching a svn diff here.

Ciao

Max

File Attachments
1) IdeImportConditions.patch, downloaded 375 times

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Tue, 04 May 2021 21:21:55 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Tue, 04 May 2021 23:11Ok... first mod is done.
With this patch ide accepts optional (flag) inside import.ext file.

I don't remember the way to contribute, so I'm attaching a svn diff here.

Ciao

Max

It is ok. It just has to wait after 2021.1 is released. Please remind me again by then...

Mirek

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Tue, 04 May 2021 21:46:26 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 04 May 2021 23:21
It is ok. It just has to wait after 2021.1 is released. Please remind me again by then...
Mirek

Ok, thanx!
I hope to have some time for other 2 patches.
Forst onw is to allow duplicate source file names in different folders:

folder1/version.cpp
folder2/version.cpp
folder2/version.c

which could be compiled as

folder1_version.cpp.o
folder2_version.cpp.o
folder2_version.c.o

For example.
The other one is to allow to reference an upper folder (now it doesn't work) :

files ../upperfolder/*.cpp

which is useful to separate a library in many nested packages.

nest
  mypackage
    mylibcode
    mypackage.upp (referencing mylibcode as mylibcode/*.cpp in ext file)
    myplugin
      myplugin.upp (referencing mylibcode as ../mylibcode/*.cpp in ext file)

I guess this could allow import of most libraries.

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Wed, 05 May 2021 12:25:26 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Tue, 04 May 2021 23:46mirek wrote on Tue, 04 May 2021 23:21
It is ok. It just has to wait after 2021.1 is released. Please remind me again by then...
Mirek

Ok, thanx!
I hope to have some time for other 2 patches.
Forst onw is to allow duplicate source file names in different folders:

folder1/version.cpp
folder2/version.cpp
folder2/version.c

which could be compiled as

folder1_version.cpp.o
folder2_version.cpp.o
folder2_version.c.o

For example.
The other one is to allow to reference an upper folder (now it doesn't work) :

files ../upperfolder/*.cpp

which is useful to separate a library in many nested packages.

nest
  mypackage
    mylibcode

mypackage.upp (referencing mylibcode as mylibcode/*.cpp in ext file)
myplugin
myplugin.upp (referencing mylibcode as ../mylibcode/*.cpp in ext file)

I guess this could allow import of most libraries.

I actually think this problem might be there even without import.ext, so it is definitely worth fixing...

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Wed, 05 May 2021 15:31:37 GMT
View Forum Message <> Reply to Message

mirek wrote on Wed, 05 May 2021 14:25

I actually think this problem might be there even without import.ext, so it is definitely worth fixing...

Agree... but now there's a problem : how shall we fix it ?

solution 1 :

/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgWidget/Util.cpp
-->
outputdir/_home_massimo_sources_OSG_OSG_OpenSceneGraph_src_osgWidget_Util.cpp.o

which I'm afraid will make windows command line limitations quite unhappy when linking the whole stuff.

solution 2 :

/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgWidget/Util.cpp
-->
outputdir/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgWidget/Util.cpp.o

probably same problem as point 1 on windows, with directory tree management plus

solution 3 :
strip all "common" path part from sources and use just the remaining as solution 1

/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgWidget/Version.cpp
/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgUtil/Version.cpp
-->
outputdir/osgWidget_Version.cpp.o
outputdir/osgUtil_Version.cpp.o

this should make windows linker happy, but it will cause complete rebuild if we add some new source file

having a different path and needs a source files preventive scan to get the common path part.

solution 4 :
some sort of file mapping

/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgWidget/Version.cpp
/home/massimo/sources/OSG/OSG/OpenSceneGraph/src/osgUtil/Version.cpp
-->
outputdir/1.o
outputdir/2.o

The map could be stored in output folder along with object files, and updated on demand.
IMHO that one would be the better solution...

what do you think about ?

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Wed, 05 May 2021 17:18:34 GMT
View Forum Message <> Reply to Message

Attached file implements solution 4 for GCCBuilder... it seems to work.
It create obj files in form of 00000000.o ... 99999999.o and stores in a map the relationship between
source file and obj file, which is recorded as OBJFILES.MAP in output folder.

If you agree with this method, I can do it also for MSC builder.

Ciao

Max

File Attachments
1) GccBuilder.diff, downloaded 338 times

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Thu, 06 May 2021 06:09:19 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Wed, 05 May 2021 17:31
which I'm afraid will make windows command line limitations quite unhappy when linking the whole stuff.

You do not have to worry about that, we are using response files (that is basically commandline in

a file) when commandline is too long. Hit that limit even as things are now.

I think that somethine like (1) is probably the best.

Mirek

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Thu, 06 May 2021 14:02:48 GMT
View Forum Message <> Reply to Message

Hi,

here the complete patch.
Tested (quickly) on Linux, so no MSC builder, I'll do it if it's all ok.

'import.ext' now accepts flags:

files(POSIX & !SOMETHING) some/path/*.cpp

allows relative paths (from current package) and absolute paths (useless, but anyways):

files(POSIX & !SOMETHING) ../myoriginallib/some/path/*.cpp
files /home/massimo/some/path/*.c

and finally allows duplicate file names in different paths and in c/c++/cxx files:

files some/path/version.cpp
files another/path/version.cpp
files another/path/version.c

object files are put in output directory with names prepended with original paths and with source extension before .o :

some/path/test.cpp --> outdir/package_full_path_some_path_test.cpp.o

Patch is attached. Please tell if it's ok and if I shall do the same for MSC too.

File Attachments

1) ide.patch, downloaded 354 times

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Sat, 15 May 2021 08:35:02 GMT
View Forum Message <> Reply to Message

Unfortunately, patch is against outdated CppBuilder.cpp (bug reported by Oblivion was fixed meanwhile), so it cannot be applied (I have actually spent some time trying to figure out whether this is not a bug in TheIDEs "Apply patch" function, but it really is not).

Before I start fixing the patch, maybe you could send me the plain CppBuilder.cpp?

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Sat, 15 May 2021 12:26:07 GMT
View Forum Message <> Reply to Message

I can try to re-fetch the three and apply my patch on updated code... just wait a little.
I changed both cppbuilder and gccbuilder (latter, iirc, for adding path to output file names).

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mdelfede on Sat, 15 May 2021 12:49:05 GMT
View Forum Message <> Reply to Message

Hi Mirek,

here you have the 2 files up-to-date with last svn checkout

(beware, msc mod is missing!)

File Attachments
1) 2021_05_15-IDE_MODS.zip, downloaded 304 times

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Mon, 17 May 2021 07:59:04 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Sat, 15 May 2021 14:49Hi Mirek,

here you have the 2 files up-to-date with last svn checkout

For some reason it is not. Check https://github.com/ultimatepp/ultimatepp/blob/de0ac2c77c48bffbccba675a367756d7e405232f/uppsrc/ide/Builders/CppBuilder.cp p#L319.

But never mind, I will manage.

Mirek

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Mon, 17 May 2021 08:08:38 GMT
View Forum Message <> Reply to Message

Error executing c:\upp\bin/clang/bin\c++.exe -c -I"C:\upp\MyApps" -I"C:\u\upp.src\uppsrc" -I"C:\upp\UppHub\eigen" -I"C:\upp\UppHub\Scatter" -I"C:\upp\UppHub\Skylark" -I"C:\upp\UppHub\URR" -I"C:\upp\UppHub\XMLMenu" -I"c:\upp\bin/SDL2/include" -I"c:\upp\bin/pgsql/x64/inc
lude" -I"c:\upp\bin/mysql/include" -I"C:/upp/out/MyApps/plugin/lzma/CLANGx64.Debug.Debug_Full.Usemalloc" -DflagUSEMALLOC -DflagCLANG -DflagDEBUG -DflagDEBUG_FULL -DflagBLITZ -DflagWIN32 -mthreads -gcodeview -fno-limit-debug-info -g2 -static -fexceptions -D_DEBUG -x c++ "C:\u\upp.src\uppsrc\plugin\lzma\lzma.cpp" -o "C:/upp/out/MyApps/plugin/lzma/CLANGx64.Debug.Debug_Full.Usemalloc\:_u_upp.src_uppsrc_plugin_lzma_lzma.cpp.o"
error: unable to open output file 'C:/upp/out/MyApps/plugin/lzma/CLANGx64.Debug.Debug_Full.Usemalloc\:_u_upp.src_uppsrc_plugin_lzma_lib_LzFind.c.o': 'Invalid argument'

Subject: Re: [PROPOSAL] A couple of changes in "import.ext"
Posted by mirek on Mon, 17 May 2021 08:35:13 GMT
View Forum Message <> Reply to Message

OK, applied with changes, please check when you have time...
