Subject: Re: Upp package binding for LLVM/Clang library (libclang)
Posted by Sender Ghost on Tue, 22 Apr 2014 22:41:35 GMT
View Forum Message <> Reply to Message

koldo wrote on Tue, 22 April 2014 12:58With your v3.3 file, llvm package compiles well after adding include folders, although clang++ emits many "'&&' within '||' [-Wlogical-op-parentheses]" warnings

It's possible to disable warnings with "-w" compiler option (or to specify "-ferror-limit=n", where n is number of diagnostics, as said in "Clang compiler user's manual").
koldo wrote on Tue, 22 April 2014 12:58Unfortunately the link fails because of "undefined reference to `_Unwind_Resume'"
Looks like, you used compiler with a different stack unwinding method: Dwarf-2 (DW2) or SJLJ (setjmp/longjmp). I used Nuwen GCC compiler (10.3 distribution version, before it became 64-bit only, but available 10.4 version might be ok) to build 3.3 version.

In case, if you used TDM-GCC 4.8.1 compiler, I rebuild 3.3 version of LLVM/Clang with it and got the same "undefined reference to `_Unwind_Resume'" errors. So, it might be compiler dependent.

I fixed issue with "unknown argument" for Clang (v3.4 release) with following changes:
Toggle patch file

```
diff -ruN llvm_3_4.orig/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td
llvm_3_4/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td
--- llvm_3_4.orig/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td 2014-01-05 02:28:17
+0400
+++ llvm_3_4/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td 2014-04-22 21:07:04
+0400
@@ -82,7 +82,7 @@

 def err_drv_I_dash_not_supported : Error<
   "'%0' not supported, please use -iquote instead">;
-def err_drv_unknown_argument : Error<"unknown argument: '%0'">;
+def err_drv_unknown_argument : Warning<"unknown argument: '%0'">;
 def err_drv_invalid_value : Error<"invalid value '%1' in '%0'">;
 def err_drv_invalid_int_value : Error<"invalid integral value '%1' in '%0'">;
 def err_drv_invalid_remap_file : Error<
diff -ruN llvm_3_4.orig/tools/clang/lib/Frontend/CompilerInvocation.cpp
llvm_3_4/tools/clang/lib/Frontend/CompilerInvocation.cpp
--- llvm_3_4.orig/tools/clang/lib/Frontend/CompilerInvocation.cpp 2014-01-05 02:25:13 +0400
+++ llvm_3_4/tools/clang/lib/Frontend/CompilerInvocation.cpp 2014-04-22 21:07:54 +0400
@@ -1644,7 +1644,7 @@
   for (arg_iterator it = Args->filtered_begin(OPT_UNKNOWN),
         ie = Args->filtered_end(); it != ie; ++it) {
     Diags.Report(diag::err_drv_unknown_argument) << (*it)->getAsString(*Args);
-    Success = false;
+    //Success = false;
   }
```

```
   Success = ParseAnalyzerArgs(*Res.getAnalyzerOpts(), *Args, Diags) && Success;
diff -ruN llvm_3_4.orig/tools/clang/tools/driver/cc1as_main.cpp
llvm_3_4/tools/clang/tools/driver/cc1as_main.cpp
--- llvm_3_4.orig/tools/clang/tools/driver/cc1as_main.cpp 2014-01-05 02:28:27 +0400
+++ llvm_3_4/tools/clang/tools/driver/cc1as_main.cpp 2014-04-22 21:08:41 +0400
@@ -167,7 +167,7 @@
   for (arg_iterator it = Args->filtered_begin(cc1asoptions::OPT_UNKNOWN),
         ie = Args->filtered_end(); it != ie; ++it) {
     Diags.Report(diag::err_drv_unknown_argument) << (*it) ->getAsString(*Args);
-    Success = false;
+    //Success = false;
   }

   // Construct the invocation.
```

which turns such kind of error to warning message and doesn't stop. But while it compiles and builds, the result executables starts with exception error. For reference, I uploaded 3.4 version (with applied patch) to the following temporary links:
LLVM/Clang v3.4 release (32-bit, compiled by TDM GCC 4.8.1, 118.8 Mb).
LLVM/Clang v3.4 release (32-bit, compiled by Nuwen GCC 4.8.1, 128 Mb).

Then I decided to apply the same changes to development version of Clang (3.5):
Toggle patch file

```
diff -ruN llvm.orig/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td
llvm/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td
--- llvm.orig/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td 2014-04-22 22:39:01 +0400
+++ llvm/tools/clang/include/clang/Basic/DiagnosticDriverKinds.td 2014-04-22 22:42:48 +0400
@@ -87,7 +87,7 @@

 def err_drv_I_dash_not_supported : Error<
   "'%0' not supported, please use -iquote instead">;
-def err_drv_unknown_argument : Error<"unknown argument: '%0'">;
+def err_drv_unknown_argument : Warning<"unknown argument: '%0'">;
 def err_drv_invalid_value : Error<"invalid value '%1' in '%0'">;
 def err_drv_invalid_int_value : Error<"invalid integral value '%1' in '%0'">;
 def err_drv_invalid_remap_file : Error<
diff -ruN llvm.orig/tools/clang/lib/Frontend/CompilerInvocation.cpp
llvm/tools/clang/lib/Frontend/CompilerInvocation.cpp
--- llvm.orig/tools/clang/lib/Frontend/CompilerInvocation.cpp 2014-04-22 22:39:06 +0400
+++ llvm/tools/clang/lib/Frontend/CompilerInvocation.cpp 2014-04-22 22:43:25 +0400
@@ -1708,7 +1708,7 @@
   for (arg_iterator it = Args->filtered_begin(OPT_UNKNOWN),
         ie = Args->filtered_end(); it != ie; ++it) {
     Diags.Report(diag::err_drv_unknown_argument) << (*it)->getAsString(*Args);
-    Success = false;
+    //Success = false;
   }
```

```
   Success = ParseAnalyzerArgs(*Res.getAnalyzerOpts(), *Args, Diags) && Success;
diff -ruN llvm.orig/tools/clang/tools/driver/cc1as_main.cpp
llvm/tools/clang/tools/driver/cc1as_main.cpp
--- llvm.orig/tools/clang/tools/driver/cc1as_main.cpp 2014-04-22 22:39:41 +0400
+++ llvm/tools/clang/tools/driver/cc1as_main.cpp 2014-04-22 22:43:41 +0400
@@ -168,7 +168,7 @@
   for (arg_iterator it = Args->filtered_begin(cc1asoptions::OPT_UNKNOWN),
         ie = Args->filtered_end(); it != ie; ++it) {
     Diags.Report(diag::err_drv_unknown_argument) << (*it) ->getAsString(*Args);
-    Success = false;
+    //Success = false;
   }

   // Construct the invocation.
```

The simple console applications starts without exception errors, but others (which compiles by 3.3, 3.4 versions) is not compiled. For reference, I uploaded 3.5 development version (with applied patch) to the following temporary link:
LLVM/Clang v3.5 devel (32-bit, compiled by Nuwen GCC 4.8.1, 137.5 Mb).