

---

Subject: Crash on "package organizer" entry  
Posted by [Zbych](#) on Thu, 04 Jan 2018 19:41:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

I don't know if there is something wrong with my configuration, but recent version of TheIDE (11638) crashes every time I try to enter "Package organizer" menu:

Stack trace:

```
20:04:19:466 INFO GuiMainFn_(): Version: 11638-xenial-amd64-nogtk (64 bit) (GCC) (C++11)
(Gtk) Compiled: 01/04/2018 20:01:54
***** ASSERT FAILED: Assertion failed in /home/zbych/upp/uppsrc/Core/Value.cpp,
line 339
type < 0x8000000
```

Stack trace:

```
Upp::AddStackTrace(char*, int)
Upp::AssertFailed(char const*, int, char const*)
Upp::Value::Serialize(Upp::Stream&)
Upp::Stream& Upp::operator%<Upp::Value>(Upp::Stream&, Upp::Value&)
Upp::Ctrl::Serialize(Upp::Stream&)
Upp::Stream& Upp::operator%<UppList>(Upp::Stream&, UppList&)
PackageEditor::Serialize(Upp::Stream&)
EditPackages(char const*, char const*, Upp::String&)
Ide::EditWorkspace()
Upp::CallbackN<> Upp::callback<Ide, Ide>(Ide*, void (Ide::*)()):{lambda()#1}::operator()() const
Upp::Function<void ()>::Wrapper<Upp::CallbackN<> Upp::callback<Ide, Ide>(Ide*, void
(Ide::*)()):{lambda()#1}>::Execute()
Upp::Function<void ()>::operator()() const
Upp::Function<void ()>::Wrapper2<Upp::Function<void ()> >::Execute()
Upp::Function<void ()>::operator()() const
Upp::Ctrl::Action()
Upp::MenuItem::LeftUp(Upp::Point_<int>, unsigned int)
Upp::Ctrl::MouseEvent(int, Upp::Point_<int>, int, unsigned int)
Upp::Ctrl::MouseEvent0(int, Upp::Point_<int>, int, unsigned int)
Upp::Ctrl::MouseEventH(int, Upp::Point_<int>, int, unsigned int)
Upp::Ctrl::MEvent0(int, Upp::Point_<int>, int)
Upp::Ctrl::DispatchMouseEvent(int, Upp::Point_<int>, int)
Upp::Ctrl::DispatchMouseEvent(int, Upp::Point_<int>, int)
Upp::Ctrl::DispatchMouseEvent(int, Upp::Point_<int>, int)
Upp::Ctrl::DispatchMouse(int, Upp::Point_<int>, int)
Upp::Ctrl::DispatchMouseIn(int, int)
```

TheIDE fails this assertion:

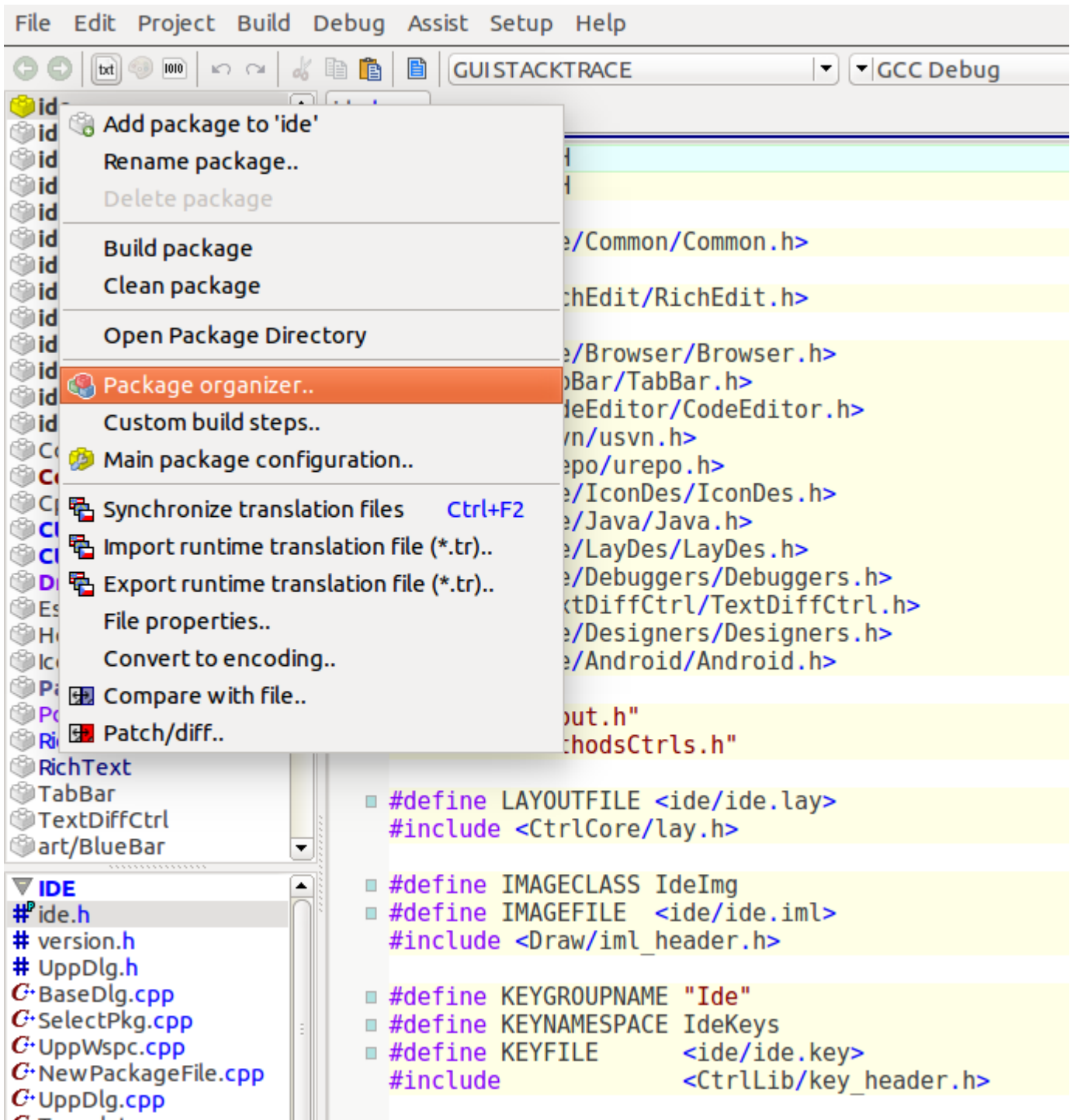
```
void Value::Serialize(Stream& s) {
    RegisterStd();
    dword type;
    if(s.IsLoading()) {
        s / type;
        ASSERT(type < 0x8000000); // only Values with assigned real type ID can be serialized
        ~~~~~~
    }
    Free();
    int st = type == VOID_V ? VOIDV : type == STRING_V ? STRING : type;
    if(st < 0)
        s.LoadError();
}
```

Correct me if I am wrong, but this assertion is not necessary and "if(st < 0)" correctly handles case when there is no more data in the stream.

### File Attachments

---

- 1) [Zrzut ekranu z 2018-01-04 20-04-44.png](#), downloaded 702 times



---

Subject: Re: Crash on "package organizer" entry  
Posted by [Zbych](#) on Thu, 04 Jan 2018 19:56:15 GMT  
[View Forum Message](#) <> [Reply to Message](#)

I guess that it is the same bug as in [https://www.ultimatepp.org/forums/index.php?t=msg&th=10173&goto=48976&#msg\\_48976](https://www.ultimatepp.org/forums/index.php?t=msg&th=10173&goto=48976&#msg_48976)

---

Subject: Re: Crash on "package organizer" entry  
Posted by [mirek](#) on Mon, 15 Jan 2018 11:16:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, I think that the correct should be:

```
void Value::Serialize(Stream& s) {
    RegisterStd();
    dword type;
    if(s.IsLoading()) {
        s / type;
        if(type >= 0x8000000)
            s.LoadError();
        Free();
        int st = type == VOID_V ? VOIDV : type == STRING_V ? STRING : type;
        if(st == STRING)
            s % data;
    }
}
```

But that is fundametaly equivalent to current code in release AFAIK.

Now the real problem here seems to be that UppList->Serialize is called (which is Ctrl::Serialize) instead of UppList->SerializeSettings.

It looks like some stuck configuration of Upp:

```
void PackageEditor::Serialize(Stream& s) {
    int version = 3;
    s / version;
    if(version >= 3) {
        filelist.SerializeSettings(s);
        package.SerializeSettings(s);
    }
    else {
        s % filelist;
        s % package % package;
    }
    SerializePlacement(s);
    if(version >= 1 && version <= 2) {
        Splitter dummy;
        s % dummy % dummy % dummy % dummy % dummy % dummy;
    }
}
```

- we should be at version 3 now, calling UppList->SerializeSettings.

Anyway, it should not crash either. If you have it in debugger, on what line exactly in Value::Serialize does it crash?

(If it is ASSERT line, does the crash happens on debug mode only?)

Mirek

---

Subject: Re: Crash on "package organizer" entry  
Posted by [Zbych](#) on Mon, 15 Jan 2018 12:15:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 15 January 2018 12:16 If you have it in debugger, on what line exactly in Value::Serialize does it crash?  
(If it is ASSERT line, does the crash happens on debug mode only?)

Yes it crashes on line ASSERT(type < 0x8000000); and only in debug mode.

So now I should ask Dolik to update his build scripts, to get rid of this assertion from ubuntu packages.

#### File Attachments

1) [1.png](#), downloaded 630 times



ide/MacroManager  
 ide/SrcUpdater  
 CodeEditor  
**Core**  
 CppBase  
 CtrlCore

BlockStream.cpp  
 FileMapping.cpp  
 # FilterStream.h  
 FilterStream.cpp  
 # Profile.h  
 # Diag.h  
 Log.cpp  
 Debug.cpp  
 # Util.h  
 Ini.cpp  
 Util.cpp  
 mathutil.cpp  
 Random.cpp  
 Speller.cpp  
 # LocalProcess.h  
 LocalProcess.cpp  
**Containers**  
 # Topt.h  
 # Range.h  
 # Algo.h  
 # CoAlgo.h  
 # Sorted.h  
 # Sort.h  
 # CoSort.h  
 # Obsolete.h  
 # Vcont.h  
 # BiCont.h  
 # Other.h  
 # Vcont.hpp  
 Vcont.cpp  
 # Index.h  
 # Map.h  
 # FixedMap.h  
 # Map.hpp  
 Hash.cpp  
 # InVector.h  
 # InVector.hpp  
 # InMap.hpp  
 # Tuple.h  
**Function**  
 # Function.h  
 CallbackNP.i  
 CallbackN.i  
 CallbackR.i  
 # Callback.h  
**Concretes**  
 # TimeDate.h  
 TimeDate.cpp  
 # Value.h

```

ide.h  GtkApp.cpp  Value.cpp  UppDlg.cpp  Stream.h
320     Value::AddName(BOOL_V, "bool");
321     Value::AddName(ERROR_V, "error");
322     };
323 }
324
325 void ValueRegisterHelper()
326 {
327     Value::RegisterStd();
328 }
329
330 INITBLOCK {
331     ValueRegisterHelper();
332 }
333
334 void Value::Serialize(Stream& s) {
335     RegisterStd();
336     dword type;
337     if(s.IsLoading()) {
338         s / type;
339     ASSERT(type < 0x80000000); // only Values with assi
340     Free();
341     int st = type == VOID_V ? VOIDV : type == STRING_V
342     if(st < 0)
343         s.LoadError();
344     else
345     if(st == STRING)
346         s % data;

```

Autos	Locals	Watches	this	CPU	Thread 1
s					{ _vptr = .Stream = 0xe474f0, pos = 0, buffer :
s.IsLoading					{ bool = (const Upp::Stream * const)
type					4294967295

Subject: Re: Crash on "package organizer" entry  
Posted by [mirek](#) on Mon, 15 Jan 2018 12:35:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Zbych wrote on Mon, 15 January 2018 13:15mirek wrote on Mon, 15 January 2018 12:16If you have it in debugger, on what line exactly in Value::Serialize does it crash?  
(If it is ASSERT line, does the crash happens on debug mode only?)

Yes it crashes on line ASSERT(type < 0x8000000); and only in debug mode.

So now I should ask Dolik to update his build scripts, to get rid of this assertion from ubuntu packages.

Well, now that would actually mean that it builds theide in debug mode, strange...

Have you tried with tarball?

---

Subject: Re: Crash on "package organizer" entry  
Posted by [Zbych](#) on Mon, 15 Jan 2018 21:52:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 15 January 2018 13:35Well, now that would actually mean that it builds theide in debug mode, strange...  
Well, I received response from Honza, that debug version is made on purpose, to let users get debug symbols by installing one extra deb package.

mirek wrote on Mon, 15 January 2018 13:35Have you tried with tarball?  
Since the assertion is gone, I will stick with ubuntu package. It is just more convenient.

Thanks for your help.

---