Subject: 2021.1 (alpha)
Posted by mirek on Sun, 21 Mar 2021 23:34:41 GMT
View Forum Message <> Reply to Message

I think it is about time to do next release. Consider current nightly as "alpha":

2021.1 (April 2021?)

Upcoming release

Release highlights
* New system for 3rd party modules, "UppHub": independently developed U++ modules residing in Git repositories now can be seamlessly integrated, including automatic installation of missing packages if they are in the UppHub.

Core
* double NAN and INF values are now considered Null (IsNull returns true for them).
* FileSystemInfo now returns volume names of network drives
* Improvements in FindFile
* String middle tier heap handling optimised
* U++ does not require MemoryFreeThread call at the end of thread (it is now implemented using thread_local destructor)

CtrlCore
* New C++ style for iteration of widget children now possible for(Ctrl& q : widget)+
* Pen support (pressure, tilt etc...)

RichText, RichEdit
* 1.15 line spacing support
* JPEG raw image data is now exported to PDF as JPEG (instead of recoding)

TheIDE
* Context goto now support AK_ key definitions
* TheIDE now contains credentials manager for git and svn (including github tokens)
* Various improvements to text comparison dialogs - ability to copy and delete individual lines between versions, filtering by file time (for directory comparison).
* Repo history of .upp files directly available via package context menu
* Open package dialog filtering options redesigned
* New function to copy positions as text to clipboard (e.g. "RichEdit/Modify.cpp:27") and go to such position based on clipboard content. Useful for communication between developers.
* Some common nest root files like README.md or LICENSE are now directly accessible through <meta> pseudo-package.
TheIDE/umk build flags MAKE_LIB and MAKE_MLIB force generation of library instead of executable.
* New "Merge nests" function (in Package organizer) copies all used packages that are not in the first nest of assembly into the first nest. In U++, first nest usually contains the application code whole while other nests contain libraries (U++ or 3rd party libraries). This function is intended to "freeze" the non-application code before releasing / tagging / branching to reduce the chance that

a change of that code does not break the application.
* TheIDE now can "Open terminal" at various directories (output directory, package directory, source file directory etc...), with PATH augmented with current build executable directories.
* Support for .md (markdown) format
* Improved debugger threads tab
* Support for override in virtual methods dialog

Win32
* Now ships with minimal git (mingit) implementation
* CLANG now at version 11.0.0

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Mon, 22 Mar 2021 07:46:29 GMT
View Forum Message <> Reply to Message

Hello Mirek,

Thanks for all your efforts!

Another important change to be logged:

* Core/SSH: the underlying libssh2 library is upgraded to v. 1.9.0

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Thu, 22 Apr 2021 09:51:13 GMT
View Forum Message <> Reply to Message

Hi!

It's an impressive list of improvements again.

How does the release schedule look like?

Best regards,

Tom

Subject: Re: 2021.1 (alpha)
Posted by mirek on Thu, 22 Apr 2021 16:52:18 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Thu, 22 April 2021 11:51Hi!

It's an impressive list of improvements again.

How does the release schedule look like?

Best regards,

Tom

I welcome suggestions...

Mirek

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Fri, 23 Apr 2021 07:14:17 GMT
View Forum Message <> Reply to Message

Hi Mirek,

If there are not any show stoppers (i.e. pending issues or features required to make it to this release), why not start the final testing stage now?

Although I do follow the development most of the time, I only use stable U++ releases for my products. This is the only way for me to know that the development is not in the middle of something critical causing some potential bug end up in my product and result in a bunch of user support cases.

Best regards,

Tom

Subject: Re: 2021.1 (alpha)
Posted by mirek on Fri, 23 Apr 2021 07:24:00 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Fri, 23 April 2021 09:14Hi Mirek,

If there are not any show stoppers (i.e. pending issues or features required to make it to this release), why not start the final testing stage now?

Although I do follow the development most of the time, I only use stable U++ releases for my

products. This is the only way for me to know that the development is not in the middle of something critical causing some potential bug end up in my product and result in a bunch of user support cases.

Best regards,

Tom

Well, I thought we are in final testing phase now.

If anybody wants to help, the most ugly part of testing is to test all those distros and exotic systems from the clean slate (like cleanly installed system).

That said, one thing that should probably still be added / fixed is pen history in gtk. I hope to add that tomorrow...

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Fri, 23 Apr 2021 08:41:09 GMT
View Forum Message <> Reply to Message

Hi,

In that case I will do a full fresh install of the latest nightly and compile all my production apps with it.

I can also do a fresh Linux Mint 20.1 installation and bring in the latest nightly there for testing.

I guess, it's worth putting out a release candidate announcement to get more people involved to test their stuff now.

Best regards,

Tom

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Fri, 23 Apr 2021 09:08:54 GMT
View Forum Message <> Reply to Message

Hi,

I'm missing the "Project > Repo > Synchronize C:\upp\upp.src\uppsrc" which was available in 15260. At this time I only have the option to synchronize the current package, e.g. "Synchronize C:\upp\upp.src\uppsrc\Core" or alternatively everything, including my own nest. Would it be much trouble to add separate "Synchronize" -lines for each of the nests in use?

Best regards,

Tom

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Fri, 23 Apr 2021 09:14:07 GMT
View Forum Message <> Reply to Message

Hi,

In Windows all my products seem to compile and run. I also updated MSBT19 / MSBT19x64 to its latest version yesterday.

Best regards,

Tom

EDIT: More testing: Fresh Linux Mint 20.1 installation with latest nightly easily compiles, installs and runs. Installing SVN sources from the menu additionally required:
sudo apt-get install subversionto run.

Subject: Re: 2021.1 (alpha)
Posted by mirek on Sat, 24 Apr 2021 07:02:20 GMT
View Forum Message <> Reply to Message

Another important issue: Ubuntu 2021.4 is supposed to ship with Wayland default. We need to check U++ works with it, with fresh installation, ideally on real HW.

Mirek

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 24 Apr 2021 07:21:17 GMT
View Forum Message <> Reply to Message

Quote:
Ubuntu 2021.4 is supposed to ship with Wayland default. We need to check U++ works with it, with fresh installation, ideally on real HW.

U++ with GNOME/Wayland works just fine -via xwayland- at least on other distros (Arch/Fedora and their derivatives). I'm using U++ on wayland daily on two of my machines since ~2019.2.

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by Tom1 on Sat, 24 Apr 2021 09:07:14 GMT
View Forum Message <> Reply to Message

Hi,

I did a fresh install of Ubuntu 2021.4 on VMWare Workstation Player. It seems to be running a process called Xwayland, so I suppose it is on wayland then.

Last nightly U++ compiled, installed and runs well and I can also compile and run UWord without trouble therein.

Best regards,

Tom

Subject: Re: 2021.1 (alpha)
Posted by mirek on Sat, 24 Apr 2021 09:08:52 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Sat, 24 April 2021 09:21Quote:
Ubuntu 2021.4 is supposed to ship with Wayland default. We need to check U++ works with it, with fresh installation, ideally on real HW.

-via xwayland-

I do not know technical details details; I wonder

- why is it "via xwayland" when we are using gtk/gdk as backend? Somehow I have hoped the using gtk as backend will solve most of problems for us.

- what are the practical implication? How is it with other software? Is running "via xwayland" standard?

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 24 Apr 2021 09:23:29 GMT
View Forum Message <> Reply to Message

Quote:- why is it "via xwayland" when we are using gtk/gdk as backend? Somehow I have hoped the using gtk as backend will solve most of problems for us.

- what are the practical implication? How is it with other software? Is running "via xwayland" standard?

It solves windowing problems but IIRC it did not solve input problems. See the workaround I sent some time ago CtrlCore/GtkApp.cpp:57

```
#if GTK_CHECK_VERSION(3, 10, 0)
 gdk_set_allowed_backends("x11"); // this fixes wayland issues
#endif
```

If you remove this, we don't get input, IIRC.

Most of the existing software -sans official and new GNOME/KDE apps- are running over xwayland if wayland is enabled. And xwayland is installed on virtually all GNOME/KDE distros, because without it you can't run most of the production software that uses X. Hence no backward compatibility. At All. Xwayland will be with us for a long time Still, it would be better if we solve the input problem and move to full wayland.

Link to official gnome page for xwayland: https://wiki.gnome.org/Initiatives/Wayland/XWayland

It is an official middleware to make the transition smooth (in long term). It does not have a real deal-breaker downside, IMO.

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 24 Apr 2021 09:39:59 GMT
View Forum Message <> Reply to Message

I'll check the workaround to see if anything is improved.. (After all, it was two releases ago.)

The problem I described was this: https://www.ultimatepp.org/forums/index.php?t=msg&th=10937&goto=53171&#msg_53171

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 24 Apr 2021 11:44:36 GMT
View Forum Message <> Reply to Message

Well,

Pure wayland:

Good news: input mostly works (menu item selection has some problems like the mouse pointer selection point is offsetted)

Bad news : No window decoration. wayland requires csd (client side decoration) GTK can draw them for us, maybe using the header bar, but it is not enabled. So ATM we have no window title, etc..

Best regards,
Oblivion.

.

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 24 Apr 2021 18:27:35 GMT
View Forum Message <> Reply to Message

Hi,

There are two relevant functions in GDK:

gdk_wayland_window_announce_ssd(GdkWindow *)
gdk_wayland_window_announce_csd(GdkWindow *)

I think the former should be explicitly called on wayland when a window with any kind of decoration is created, unless we want csd, which seems to be the default settings for some reason.

A link to function prototypes: https://gitlab.gnome.org/GNOME/gtk/-/blob/gtk-3-24/gdk/wayland/gdkwaylandwindow.h
A link that explains how to detect wayland backend:
https://developer.gnome.org/gdk3/stable/gdk3-Wayland-Interac tion.html

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by pvictor on Wed, 28 Apr 2021 08:43:26 GMT
View Forum Message <> Reply to Message

Hi,

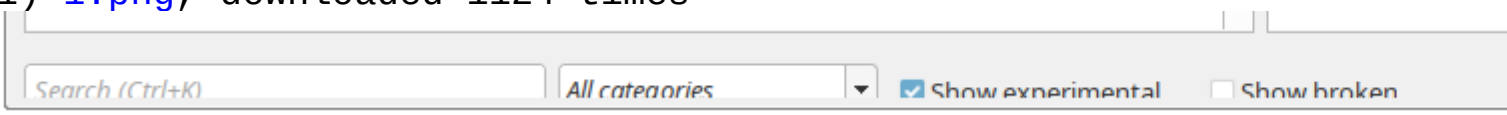
Here's how the bottom of UppHub window looks sometimes:

But sometimes, after reloading TheIDE, it looks fine.
All other windows are fine allways.
The current nightly (15935) for Linux.

Best regards,
Victor

File Attachments

1) 1.png, downloaded 1124 times



Subject: Re: 2021.1 (alpha)
Posted by Klugier on Wed, 28 Apr 2021 16:56:43 GMT
View Forum Message <> Reply to Message

Hello Victor,

Thanks for the report! Should be fixed in tomorrows nightly build (29.04.2021).

Klugier

Subject: Re: 2021.1 (alpha)
Posted by mirek on Thu, 29 Apr 2021 06:39:17 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Sat, 24 April 2021 11:23

It is an official middleware to make the transition smooth (in long term). It does not have a real deal-breaker downside, IMO.

Best regards,
Oblivion

BTW, does my recent fix for debugger/capture issue work in xwayland?

(It is using "setxkbmap -option grab:break_action" to release any capture held by the application)

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Thu, 29 Apr 2021 10:56:10 GMT
View Forum Message <> Reply to Message

Quote:does my recent fix for debugger/capture issue work in xwayland?

Yes, it works fine, here.

Edit: A correction. Grab works fine, but I have just noticed that either this fix or something else messes up the keyboard input for ALL U++ applications on wayland from that point on, if I stop the debugging. (Reverts the keyboard to US ASCII and his only happens for U++ apps.) Way to restore the kbd is to log out or restart the xwayland.

Edit 2: Ok, I can confirm that this fix breaks the keyboard input.

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 01 May 2021 07:59:18 GMT
View Forum Message <> Reply to Message

Hello Mirek,

There seem to be more bad news:Gnome does not, and apparently for the foreseeable future, will not, support the server side window decorations on wayland. They delegated that task to the client code.So it appears that we have to build the decorations ourselves, using what gnome provides: HeaderBars, etc.

Best regards,
Oblivion

Subject: Re: 2021.1 (alpha)
Posted by mirek on Sat, 01 May 2021 08:21:38 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Thu, 29 April 2021 12:56Quote:does my recent fix for debugger/capture issue work in xwayland?

Yes, it works fine, here.

Edit: A correction. Grab works fine, but I have just noticed that either this fix or something else messes up the keyboard input for ALL U++ applications on wayland from that point on, if I stop the

debugging. (Reverts the keyboard to US ASCII and his only happens for U++ apps.) Way to restore the kbd is to log out or restart the xwayland.

Edit 2: Ok, I can confirm that this fix breaks the keyboard input.

Best regards,
Oblivion

Any suggestions? IMO this is bad but sort of acceptable... Or maybe it is possible to switch something back at the end of debug session?

Mirek

Subject: Re: 2021.1 (alpha)
Posted by Oblivion on Sat, 01 May 2021 08:45:02 GMT
View Forum Message <> Reply to Message

Quote:Any suggestions? IMO this is bad but sort of acceptable... Or maybe it is possible to switch something back at the end of debug session?

Well, this turned out to be a problem on my side (or on GNOME with default wayland, where X11 conf in /etc is not really configured.): The keymap for x11 was unset. so I have "fixed" this by using "localectl" to create the keymap conf in etc with my preferred keymap (tr,tr). Now it works as expected.

Best regards,
Oblivion
