Subject: Improving the organization of U++ help
Posted by gprentice on Sat, 30 May 2009 13:39:34 GMT
View Forum Message <> Reply to Message

Here's what I think a properly organised U++ "help" would look like. It's very likely to be incomplete and I'm not offering to do any of the work (right now)!

1. Introduction
1.1 Overview of U++. (no major details)
U++ is C++ development platform for creating console or GUI applications
- written entirely in C++ using modern techniques.

Platforms supported.
Compilers and compiler versions supported.
U++ IDE is called "theIDE" and is built with U++.
Includes comprehensive editor, build system and project management.
Design philosophy - emphasis is on "rapid development"? What about runtime performance?
How does "objects are on the stack" philosophy affect code size?
Look and feel / chameleon
Visual designer
Database capability
Multithreading issues
NTL ??
Link to list of all widgets - what widgets are missing.
Image designer.
Core values and containers - what are they? - link to tutorials.
Non GUI libraries (important ones) - SQL, Web etc - link to packages list
Assist++ - sophisticated source navigation.
Topic++
Blitz & uld.

1.2 Using help. What is "documents", "reference", "implementation".
What generates the stuff in the help tree control (what is "used packages",
what is "other packages", is this list context dependent)
Link to "topic++" topic and explain better how source code is linked to documentation. How to search help.

1.3 Creating a simple project from scratch - say, the button example, and run it.
(include setting up the compiler and the build mode).

1.4 Library orginization.
What is "core". what is ctrllib. what is ctrlcore etc.

1.5 Project organization
Summarise packages and assemblies.

1.6 List of tutorials

1.7 Why U++ exists, who is using it.

1.8 Obtaining and installing U++
include how to get updates from google code.

1.9 Deploying U++ apps.

1.10 Examples & references
Explain about them.

2. Using theIDE
    2.1 Describe main components of theIDE UI.
        Explain what all the toolbars do (including "aux" thing) and package lists.
        How to select a project (describe select package dialog).
        Describe all menu options.
        Describe how to turn off an on all toolbars
            (why is there no "view" menu and no close button on the toolbars?)
    2.2 Keyboard shortcuts
    2.2 Using the editor.
            Mention auto-saving.

    2.3 Layout designer
    2.4 Image designer
    2.5 theIDE command line & UMK
    2.6 cross development topic
    2.7 File types

3. Project/package organization & configuration & building & debugging
    3.1 Packages, assemblies & nests
    3.2 Configuring packages, assemblies
    3.3 Setting compiler options & build flags
    3.4 Building - where does output go. Can I edit while building?
    3.5 Package templates
    3.6 Debugging

4. Creating a GUI application
    - GUI tutorial
5. Widgets

6. ESC scripting
    - include "theIDE macros" topic from "theIDE help"

7. Library/ API reference & docs

Core, ctrlcore etc

8. Extending U++.

--
Graeme

Subject: Re: Improving the organization of U++ help
Posted by mirek on Sat, 30 May 2009 20:39:19 GMT

View Forum Message <> Reply to Message

gprentice wrote on Sat, 30 May 2009 09:39
Here's what I think a properly organised U++ "help" would look like. It's very likely to be incomplete and I'm not offering to do any of the work (right now)!

1. Introduction
1.1 Overview of U++. (no major details)
U++ is C++ development platform for creating console or GUI applications
- written entirely in C++ using modern techniques.

Platforms supported.
Compilers and compiler versions supported.
U++ IDE is called "theIDE" and is built with U++.
Includes comprehensive editor, build system and project management.
Design philosophy - emphasis is on "rapid development"? What about runtime performance?
How does "objects are on the stack" philosophy affect code size?
Look and feel / chameleon
Visual designer
Database capability
Multithreading issues
NTL ??
Link to list of all widgets - what widgets are missing.
Image designer.
Core values and containers - what are they? - link to tutorials.
Non GUI libraries (important ones) - SQL, Web etc - link to packages list
Assist++ - sophisticated source navigation.
Topic++
Blitz & uld.

1.2 Using help. What is "documents", "reference", "implementation".
What generates the stuff in the help tree control (what is "used packages",
what is "other packages", is this list context dependent)
Link to "topic++" topic and explain better how source code is linked to documentation.   How to search help.

1.3 Creating a simple project from scratch - say, the button example, and run it.
(include setting up the compiler and the build mode).

1.4 Library orginization.
What is "core". what is ctrllib. what is ctrlcore etc.

1.5 Project organization
Summarise packages and assemblies.

1.6 List of tutorials

1.7 Why U++ exists, who is using it.

1.8 Obtaining and installing U++
include how to get updates from google code.

1.9 Deploying U++ apps.

1.10 Examples & references
Explain about them.

2. Using theIDE
- 2.1 Describe main components of theIDE UI.
  - Explain what all the toolbars do (including "aux" thing) and package lists.
  - How to select a project (describe select package dialog).
  - Describe all menu options.
  - Describe how to turn off an on all toolbars
    - (why is there no "view" menu and no close button on the toolbars?)
- 2.2 Keyboard shortcuts
- 2.2 Using the editor.
    - Mention auto-saving.

- 2.3 Layout designer
- 2.4 Image designer
- 2.5 theIDE command line & UMK
- 2.6 cross development topic
- 2.7 File types

3. Project/package organization & configuration & building & debugging
- 3.1 Packages, assemblies & nests
- 3.2 Configuring packages, assemblies
- 3.3 Setting compiler options & build flags
- 3.4 Building - where does output go. Can I edit while building?
- 3.5 Package templates
- 3.6 Debugging

4. Creating a GUI application
    - GUI tutorial
5. Widgets

6. ESC scripting
    - include "theIDE macros" topic from "theIDE help"

7. Library/ API reference & docs
   Core, ctrlcore etc

8. Extending U++.

--
Graeme

I believe this is a good idea & start. We should build on this.

Just, please, do not expect all docs to be written by me. I believe I can do part of reference docs and most library tutorials.

Mirek

Subject: Re: Improving the organization of U++ help
Posted by koldo on Sat, 30 May 2009 22:04:18 GMT

View Forum Message <> Reply to Message

Quote:I believe this is a good idea & start. We should build on this.

Just, please, do not expect all docs to be written by me. I believe I can do part of reference docs and most library tutorials.

Hello all

That's it. I great project needs a good documentation.

Many documents are around there. This index could help to give all U++ documentation a common organization to be got by clicking F1 in TheIde and also to be in one .pdf file to be distributed or just printed.

I recognize that perhaps many of us like to program but do not like so much to document. And also a good team has to have good coders, but also:

- good software engineers to design the project
- good style people to design useful and nice interfaces

- good users to test the libraries finding bugs and helping to solve them

But also people to help documenting.

If some programmer do not have time or abilities to document his/her code, perhaps other person can help.

I think that for documentation and also for Upp developing it could be interesting to:

- Decide main areas to develop
- Define the structure of them. The contents of Upp manual gprentice has defined is a good example
- Detail tasks
- Put tasks in a kind of "bulletin board" for community volunteers to choose those that best fit with them

Best regards
Koldo

Subject: Re: Improving the organization of U++ help
Posted by gprentice on Sat, 30 May 2009 23:36:50 GMT

View Forum Message <> Reply to Message

> luzr wrote on Sun, 31 May 2009 08:39
> I believe this is a good idea & start. We should build on this.
>
> Just, please, do not expect all docs to be written by me. I believe I can do part of reference docs and most library tutorials.
>
> Mirek

ok. So how will it be organized? Can the existing help have two modes - one that has full index in the tree control and one that has "package restricted" index like at present. Is it possible to auto-generate a pdf of the whole lot - including indexed table of contents. If pdf is too hard or nobody has acrobat pro, then html like the website.

As koldo mentioned, for people to contribute there needs to be a way of letting people assign tasks to themselves so there's no doubling up. Also, since most people can't predict when they'll get time to work on it, maybe an expected completion date could be posted so if someone else gets keen to help, they can. Even a revision control like system so people can check out topics for updating without fear their work will conflict with someone else's.

Also, the "help outline" I posted probably needs refining. Once the table of contents is agreed to, it should be created, with skeleton pages for everything so hyper-links can be used.

Is it possible to have a search mode in the help that restricts to finding function names. You can do this in theIDE right? How do you get from the IDE list to the help topic for the function.

As well as table of contents, an index would be nice. Is there any way to insert a (hidden) marker anywhere within a topic that includes a name that would appear in the index - maybe with a category too so that the index can separate API/ function names from the rest.

There's already a lot of documentation and it might not take a lot to fill in the gaps. The focus of the website could then change slightly to have "1.1 Overview of U++" in a prominent place so people can immediately see quickly what U++ is all about.

Graeme

Subject: Re: Improving the organization of U++ help
Posted by mirek on Sun, 31 May 2009 07:10:34 GMT
View Forum Message <> Reply to Message

---

gprentice wrote on Sat, 30 May 2009 19:36luzr wrote on Sun, 31 May 2009 08:39
 I believe this is a good idea & start. We should build on this.

Just, please, do not expect all docs to be written by me. I believe I can do part of reference docs and most library tutorials.

Mirek

ok. So how will it be organized?

I guess for starters, you can use what we have, just add one T++ topic with this summary that links to existing parts, leave non-existing topics without links.

This will immediately generate it to www.

Quote:
  Can the existing help have two modes - one that has full index in the tree control and one that has "package restricted" index like at present.

It is not package restricted anymore, rather 'assembly restricted' - and that is for all purposes ok as long as you have uppsrc in the list.

Quote:
  Is it possible to auto-generate a pdf of the whole lot - including indexed table of contents. If pdf is too hard or nobody has acrobat pro, then html like the website.

With some effort, yes.

Quote:
As koldo mentioned, for people to contribute there needs to be a way of letting people assign tasks to themselves so there's no doubling up. Also, since most people can't predict when they'll get time to work on it, maybe an expected completion date could be posted so if someone else gets keen to help, they can. Even a revision control like system so people can check out topics for updating without fear their work will conflict with someone else's.

Maybe, but with current "traffic" this is now least of concerns..

Quote:
Is it possible to have a search mode in the help that restricts to finding function names. You can do this in theIDE right? How do you get from the IDE list to the help topic for the function.

Ehm, this is what "finishing" T++ was about?

Quote:
As well as table of contents, an index would be nice. Is there any way to insert a (hidden) marker anywhere within a topic that includes a name that would appear in the index - maybe with a category too so that the index can separate API/ function names from the rest.

Yes. "Index entry"

Mirek

Subject: Re: Improving the organization of U++ help
Posted by gprentice on Sun, 31 May 2009 09:39:17 GMT

View Forum Message <> Reply to Message

Quote:I guess for starters, you can use what we have, just add one T++ topic with this summary that links to existing parts, leave non-existing topics without links.

Add a topic? Where? How?

Links? You mean a hyperlink? So you can't have a topic such as "Moveable" appearing directly in both UsedPackages->Core->Documents as well as in UserManual->Library->Overview->Moveable?

Quote:
Quote:
Is it possible to auto-generate a pdf of the whole lot - including indexed table of contents. If pdf is

too hard or nobody has acrobat pro, then html like the website.

With some effort, yes.

Would it be automated?

Quote:Maybe, but with current "traffic" this is now least of concerns..

Well if documentation is better organized, traffic might improve because people will be able to see what needs doing and also will be able to more easily find the topic they have something to contribute on. If people think their work will be ignored (as some of the tutorials contributed in the past have been) or conflict with someone else, they're not gonna be interested in contributing. A visible process is needed.

Will contributions and modifications be accepted blindly without checking? Revision control would help a lot.

Quote:Ehm, this is what "finishing" T++ was about?

I've only loosely been following Upp for the last 3 years so I don't know what's been done. Do you think you could spell it out for me?

Say I put the cursor on the call to Add in button.cpp example, and press Ctrl Q. It shows two Add functions in the navigator bar. Click one and I get taken to the code. Click the little blue square box and I go to the topic++ editor for the docs for that function. How do I get to the help info for that function without being in topic++ editor mode? (yes, I know mouse-over of the little blue box gives a popup but I want to jump into the help).

Quote:Yes. "Index entry"

I see the "index entry" combo box in the topic++ editor and I notice stuff turns green where I've inserted "indexes" but I don't see any index turning up anywhere and no info about it in the help for topic++.

Another problem is if you enter a search term in the help search box, the term isn't highlighted in the topic content window and there's no way to step forward and backwards through the list of matches like most search mechanisms have.

Graeme

Subject: Re: Improving the organization of U++ help

Posted by mirek on Sun, 31 May 2009 11:16:17 GMT
View Forum Message <> Reply to Message

gprentice wrote on Sun, 31 May 2009 05:39Quote:I guess for starters, you can use what we have, just add one T++ topic with this summary that links to existing parts, leave non-existing topics without links.

Add a topic? Where? How?

Links? You mean a hyperlink? So you can't have a topic such as "Moveable" appearing directly in both UsedPackages->Core->Documents as well as in
UserManual->Library->Overview->Moveable?

What is "UserManual->Library->Overview->Moveable"?

Do you propose multiple trees?

I am afraid this would lead to requests to T++ that will take a lot of time to implement.

Meanwhile, I think that for most SW products docs, hyperlinks are enough to maintain logical structure.

Quote:
Quote:
Quote:
Is it possible to auto-generate a pdf of the whole lot - including indexed table of contents. If pdf is too hard or nobody has acrobat pro, then html like the website.

With some effort, yes.

Would it be automated?

Can be. Not harder than html->pdf conversion, obviously.

Quote:
Quote:Maybe, but with current "traffic" this is now least of concerns..

Well if documentation is better organized, traffic might improve because people will be able to see what needs doing and also will be able to more easily find the topic they have something to contribute on. If people think their work will be ignored (as some of the tutorials contributed in the past have been) or conflict with someone else, they're not gonna be interested in contributing. A visible process is needed.

Come on. svn is good enough to manage source contributions. I do not see docs any harder.

Quote:
Will contributions and modifications be accepted blindly without checking? Revision control would help a lot.

Sure, svn. All this infrastrucre exists for more than 8 months.

Quote:
Quote:Ehm, this is what "finishing" T++ was about?

I've only loosely been following Upp for the last 3 years so I don't know what's been done. Do you think you could spell it out for me?

Ah, I see. Well, what about just scanning this "Documentation" forum for details?

Quote:
How do I get to the help info for that function without being in topic++ editor mode? (yes, I know mouse-over of the little blue box gives a popup but I want to jump into the help).

Well, nobody suggested this yet. Can be added.

Quote:
I see the "index entry" combo box in the topic++ editor and I notice stuff turns green where I've inserted "indexes" but I don't see any index turning up anywhere and no info about it in the help for topic++.

Not yet implemented. Besides, in reality, full-text search is perhaps better..

Quote:
Another problem is if you enter a search term in the help search box, the term isn't highlighted in the topic content window and there's no way to step forward and backwards through the list of matches like most search mechanisms have.

Yep. Actually, help browser is still a bit lacking. From time to time, various people suggested taking a care about it, that is why I did not got involved yet...

Mirek

Subject: Re: Improving the organization of U++ help

Posted by cbpporter on Sun, 31 May 2009 11:49:56 GMT
View Forum Message <> Reply to Message

luzr wrote on Sun, 31 May 2009 14:16
Quote:
How do I get to the help info for that function without being in topic++ editor mode? (yes, I know mouse-over of the little blue box gives a popup but I want to jump into the help).

Well, nobody suggested this yet. Can be added.

If my memory serves me correctly I suggested that clicking the small green icons on the side should take you to the Topic++ page, but not the editable one like it is doing right now. I think my argument was that only the most active documentation creators would maybe want to edit the section, while average users would enjoy having instant jump to the documentation page.

Or maybe I just wanted to suggest it. I don't know, it is hard to keep track of hundreds of messages and threads. I imagine how it must be for you with thousands.

Maybe Shift+Click could take you instead to the editable version.

Subject: Re: Improving the organization of U++ help
Posted by gprentice on Sun, 31 May 2009 12:14:01 GMT
View Forum Message <> Reply to Message

Quote:What is "UserManual->Library->Overview->Moveable"?

There's a lot of (document) topics in core that I didn't make a home for in the first post in this thread. I suspect they need to be somewhere more visible than buried in the library API docs but I didn't want to go to this detail in the first post. Hence, referring to the first post in this thread, Chapter 7 is "Library", so I "added" Overview sub-section and "Moveable" sub-topic.

Quote:Do you propose multiple trees?

I am afraid this would lead to requests to T++ that will take a lot of time to implement.

Meanwhile, I think that for most SW products docs, hyperlinks are enough to maintain logical structure.

I didn't think of this problem when I first posted this thread. I think it's worth putting some thought into.

How is the existing tree populated - by scanning packages for .tpp files ? Is "theIDE help" a hard coded top level topic? Auto-updating of the help index is a feature worth keeping of course.

I notice the order of topics shown in the tree is alphabetical. There needs to be control over the order. Maybe it wouldn't be hard to have a second tree with the ability to assign a redirect option to any node in the tree that points at a .tpp file and that topic is shown directly in the new tree instead of a hyper-link. I have to think more about this.

Quote:Come on. svn is good enough to manage source contributions. I do not see docs any harder.
...
Sure, svn. All this infrastrucre exists for more than 8 months.

I'm not familiar with using svn with U++. Topic++ files are binary aren't they, and they have images in them. How does subversion handle diffing them?

An auto-generated index isn't hugely important but most "helps" have them and I tend to use them. It's not important right now.

Graeme

Subject: Re: Improving the organization of U++ help
Posted by mirek on Sun, 31 May 2009 19:57:57 GMT
View Forum Message <> Reply to Message

gprentice wrote on Sun, 31 May 2009 08:14
How is the existing tree populated - by scanning packages for .tpp files ?

Yes.

Quote:
Is "theIDE help" a hard coded top level topic? Auto-updating of the help index is a feature worth keeping of course.

Yes, it is included in .exe (so it is there even if ide package is not present).

Quote:
I'm not familiar with using svn with U++. Topic++ files are binary aren't they.

Not anymore, that was part of creating new infrastructure. Diff would work with them just OK (but of course, you would have to deal with raw QTF to solve conflicts - nothing too hard anyway).

I believe that current documentation infrastructure, while not perfect, is good enough. What is missing is content

Mirek

Subject: Re: Improving the organization of U++ help
Posted by koldo on Sun, 31 May 2009 20:54:51 GMT
View Forum Message <> Reply to Message

Hello Mirek

I am 95% agree with this statement.

Quote:I believe that current documentation infrastructure, while not perfect, is good enough. What is missing is content

There is a tremendous quantity of U++ code that is not documented, beginning just from the Summary. Also there are many fragments of documentation interesting but unlinked.

A would like to see under Help menu:

- a Summary like the proposed by gprentice

- the existing documents linked to that Summary

- an opportunity for people like me that have some time and want to help U++ project to contribute.

Just a list of documentation to be done linked to the Summary and an opportunity to sign in some place and say YES!, I will do this in four weeks!. The developer of that code only should have to check the new documentation.

If I where a developer without time to document my code and some person would offer to do that, I would be very happy.

Isn't it?

Best regards
Koldo

Subject: Re: Improving the organization of U++ help
Posted by mirek on Sun, 31 May 2009 21:02:50 GMT
View Forum Message <> Reply to Message

koldo wrote on Sun, 31 May 2009 16:54

- an opportunity for people like me that have some time and want to help U++ project to contribute.

Basically everybody with any write access to svn gains rights to edit documentation. Ditto for website.

Mirek

Subject: Re: Improving the organization of U++ help
Posted by gprentice on Mon, 01 Jun 2009 00:49:01 GMT
View Forum Message <> Reply to Message

luzr wrote

Quote:
I'm not familiar with using svn with U++. Topic++ files are binary aren't they.

Not anymore, that was part of creating new infrastructure. Diff would work with them just OK (but of course, you would have to deal with raw QTF to solve conflicts - nothing too hard anyway).

Great. (how are images handled?)

luzr wrote
I believe that current documentation infrastructure, while not perfect, is good enough. What is missing is content

What is the main content that you think is missing?

luzr wrote
Basically everybody with any write access to svn gains rights to edit documentation. Ditto for website.

Forgive my ignorance but what does "ditto for website" mean?

koldo wrote
Just a list of documentation to be done linked to the Summary and an opportunity to sign in some place and say YES!, I will do this in four weeks!. The developer of that code only should have to check the new documentation.

Google code issue tracking is a possibility for this http://code.google.com/p/upp-mirror/issues/list

People could post documentation deficiencies there and also create issues where they post updated information for the help if they don't have time to check in updated documentation. Posting a thread in (this) documentation forum would probably work too .

Graeme

Subject: Re: Improving the organization of U++ help
Posted by mirek on Mon, 01 Jun 2009 06:22:17 GMT

View Forum Message <> Reply to Message

gprentice wrote on Sun, 31 May 2009 20:49luzr wrote

Quote:
I'm not familiar with using svn with U++. Topic++ files are binary aren't they.

Not anymore, that was part of creating new infrastructure. Diff would work with them just OK (but of course, you would have to deal with raw QTF to solve conflicts - nothing too hard anyway).

Great. (how are images handled?)

Badly - as before. OTOH, given that content of images rarely changes within them, I suppose that diff would work in most cases too - these blocks would either be equal as whole or different as whole.

Sidenote: So far, for the code and documentation, revision conflicts proved to be much less concern than I originally anticipated. I mean, they do not happen all that often, even if the whole issue is unmanaged.

Quote:
luzr wrote
I believe that current documentation infrastructure, while not perfect, is good enough. What is missing is content

What is the main content that you think is missing?

I guess I am not a good one to ask. But from my point of view, I have planned following articles:

- Community manual. Started writing one in uppbox.
- Contributors manual.

- reference docs
- Sql tutorial (started this one yesterday)
- Painter tutorial
- Drag&Drop tutorial
- "Against manual resource management" article (explaining U++ programming style and typical patterns represented in it).

Quote:
luzr wrote
Basically everybody with any write access to svn gains rights to edit documentation. Ditto for website.

Forgive my ignorance but what does "ditto for website" mean?

Everybody with svn write access can alter the website. Website is generated each night from uppbox/uppweb.

Mirek