Subject: Threading example for U++
Posted by arturbac on Sun, 24 Jun 2007 22:53:20 GMT
View Forum Message <> Reply to Message

I wrote this specialy for U++.
If You find this usefull in examples place it there.

27.06.2007 old code deleted, new one atached below.

Subject: Re: Threading example for U++
Posted by mirek on Mon, 25 Jun 2007 06:47:50 GMT
View Forum Message <> Reply to Message

Thread().Priority(50);

Crashed at this line for apparent reason...

Mirek

Subject: Re: Threading example for U++
Posted by arturbac on Mon, 25 Jun 2007 07:39:40 GMT
View Forum Message <> Reply to Message

Remove the line it is not necessery for the App. On My Win it dosn't crash.

Subject: Re: Threading example for U++
Posted by mirek on Mon, 25 Jun 2007 08:00:48 GMT
View Forum Message <> Reply to Message

Actually, it was just ASSERT (thread has to opened), no problem in release mode.

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 09:30:21 GMT
View Forum Message <> Reply to Message

Maybe this one will satisfy You, it does real job.
Probaply name schould be changed from thread example to fractal explorer.

27.06.2007 old code deleted, new one atached below.

File Attachments

1) threads_ex.PNG, downloaded 1266 times



Subject: Re: Threading example for U++
Posted by unodgs on Tue, 26 Jun 2007 09:54:47 GMT
View Forum Message <> Reply to Message

Wow! Very good threading example!

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 10:22:28 GMT
View Forum Message <> Reply to Message

Thanks.

Subject: Re: Threading example for U++

Posted by ebojd on Tue, 26 Jun 2007 18:28:45 GMT
View Forum Message <> Reply to Message

I was not able to get it to build (multiple errors). Most likely problem is that I am running an incompatible version of UPP (currently version 2007.1 on gentoo linux)...

What version and OS did you develop and run on?

The code looks clean enough on first read that I vote this to be added as one of the distributed example (if appropriate, etc.)

EBo --

Subject: Re: Threading example for U++
Posted by mirek on Tue, 26 Jun 2007 18:55:05 GMT
View Forum Message <> Reply to Message

Well, I guess it will counter the efforts a bit, but I am thinking about adding it after converting it to CoWork (and perhaps improving a bit too).

Mirek

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 18:56:04 GMT
View Forum Message <> Reply to Message

I use Current, I'm not using 2007.1 at all at the moment.
Hoever i didn't check it on BSD with gcc, and will not becouse u++ team member dont want this ex. in U++ so there is no reason.

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 19:00:43 GMT
View Forum Message <> Reply to Message

luzr wrote on Tue, 26 June 2007 20:55but I am thinking about adding it after converting it to CoWork

The purpose was to show How use Threads by "hand" without CoWork, in the different way, that allows programer by hand manage entire Thread use in App.

CoWorker in U++ , BackgroundWorker in NET all are good but sometimes there are more compilated program designs wich prohibit use os such High level tools.

Subject: Re: Threading example for U++
Posted by mirek on Tue, 26 Jun 2007 19:05:41 GMT
View Forum Message <> Reply to Message

That is OK. OTOH, I guess we already have an example how to do MT applications (without CoWork) - GuiMT. This one would nicely fill the need of CoWork demonstration (which is IMO missing a bit more right now).

In fact, the needed change is quite minimal, about 10 lines are deleted and one added

Of course, it is your code, I will not alter/add without your permision

---

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 19:26:52 GMT
View Forum Message <> Reply to Message

I will think about it.

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 20:21:34 GMT
View Forum Message <> Reply to Message

With CoWork my Process() will put all cb into the CoWork queue so I will not known which are active at the moment, and this is traced in app.
So propably this is not convertible to CoWork without loosing some functionality.

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 22:40:49 GMT
View Forum Message <> Reply to Message

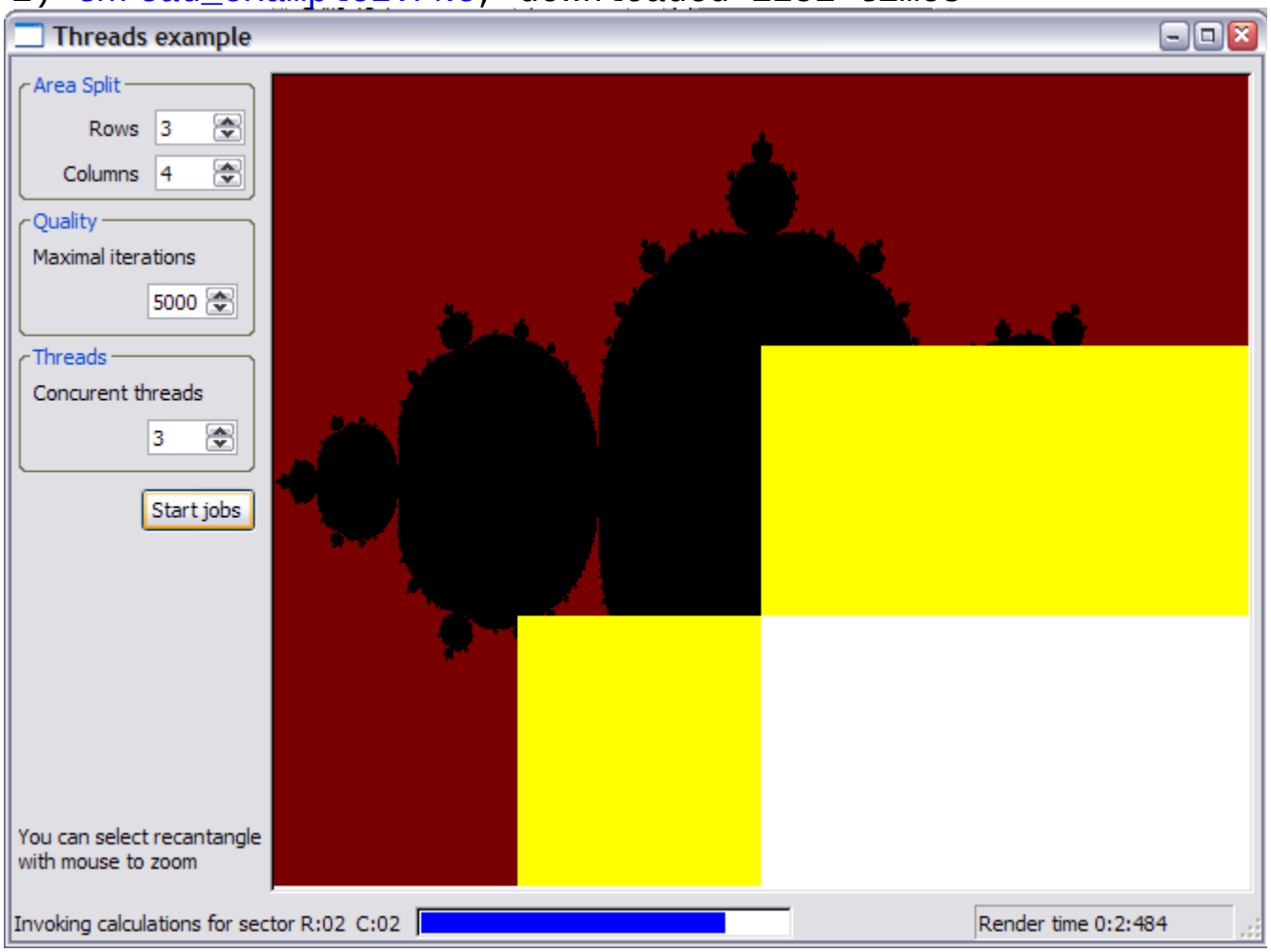
I bit corrected to look nicer but without CoWork.
Allow changing threads count, rows, cols, iterations shows time etc.

I think GuiMT is better candidate for CoWork.

Current src bellow.

File Attachments

1) thread_example2.PNG, downloaded 1251 times



Subject: Re: Threading example for U++
Posted by arturbac on Tue, 26 Jun 2007 22:43:39 GMT
View Forum Message <> Reply to Message

ebojd wrote on Tue, 26 June 2007 20:28
I was not able to get it to build (multiple errors). Most likely problem is that I am running an incompatible version of UPP (currently version 2007.1 on gentoo linux)...

Yep i checked in 2007.1 there is not Mutex (but this is no problem)

There are unresolved symbols with Semaphore and Thread classes.

in current upp there are no problems.

Subject: Re: Threading example for U++
Posted by mirek on Wed, 27 Jun 2007 06:21:57 GMT
View Forum Message <> Reply to Message

arturbac wrote on Tue, 26 June 2007 18:40
I think GuiMT is better candidate for CoWork.

Well, it is not.

CoWork is intended as a tool for simple loop paralelization. There is no loop in GuiMT....

Subject: Re: Threading example for U++
Posted by mirek on Wed, 27 Jun 2007 06:24:41 GMT
View Forum Message <> Reply to Message

ebojd wrote on Tue, 26 June 2007 14:28
I was not able to get it to build (multiple errors). Most likely problem is that I am running an incompatible version of UPP (currently version 2007.1 on gentoo linux)...

Sorry, MT was the focus for the next release, there is a lot of new stuff in the "current" (and Artur is using it

Mirek

Subject: Re: Threading example for U++
Posted by arturbac on Wed, 27 Jun 2007 09:16:28 GMT
View Forum Message <> Reply to Message

Few corrections needed for WindowsXP (not needed on XP64 lol)

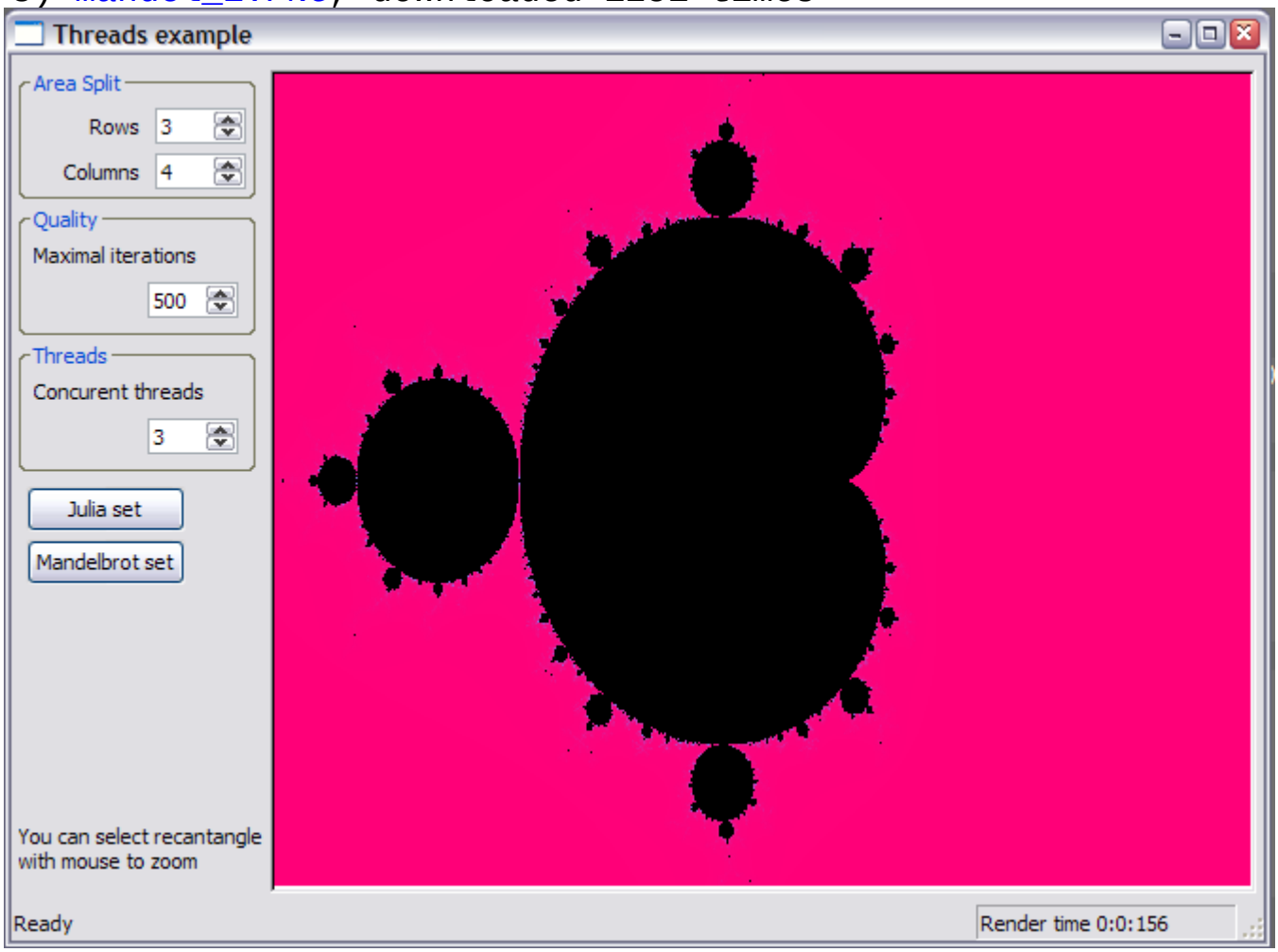
File Attachments
1) Threadsv2a.7z, downloaded 540 times

Subject: Re: Threading example for U++
Posted by arturbac on Thu, 28 Jun 2007 13:47:19 GMT
View Forum Message <> Reply to Message

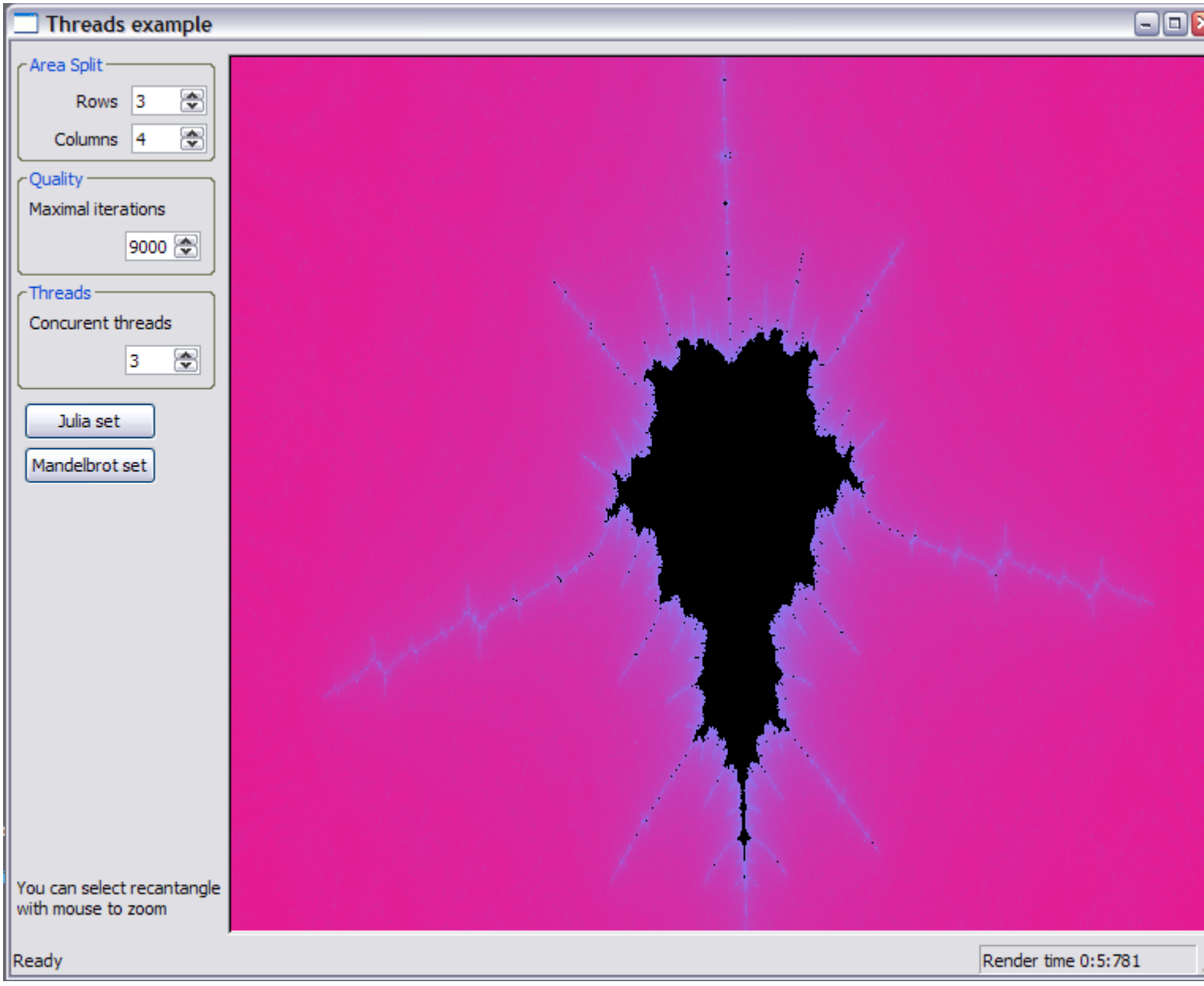
I extended application by adding possibility to process different jobs with one forking method.
So now we have Julia % Mandelbrot sets in Threads App.

File Attachments

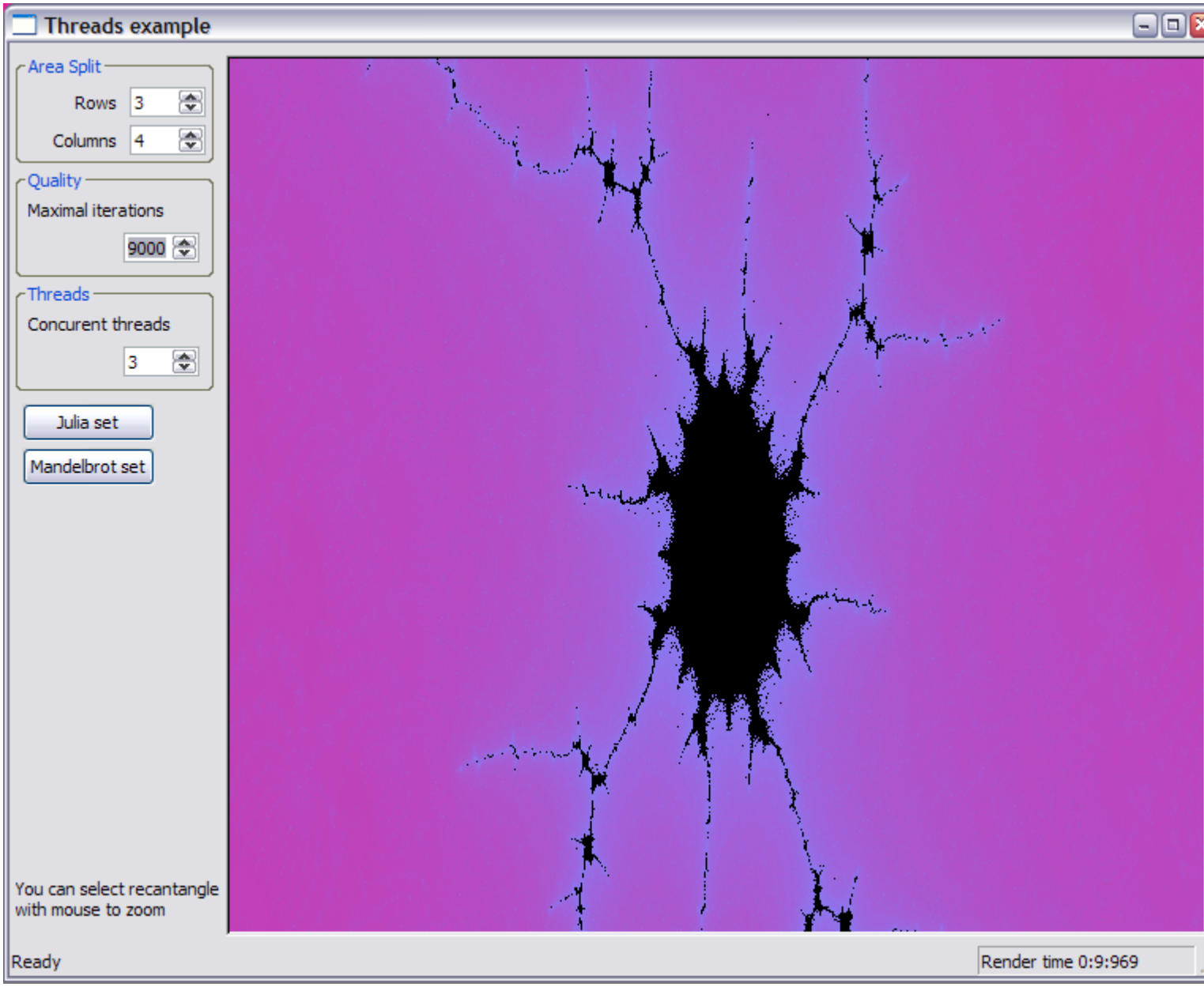
1) Threads_v3_src.7z, downloaded 601 times
2) Threads_v3_bin.7z, downloaded 541 times
3) mandel_1.PNG, downloaded 1251 times



4) mandel_3.PNG, downloaded 1265 times



5) mandel_4.PNG, downloaded 1252 times



Subject: Re: Threading example for U++
Posted by mirek on Thu, 28 Jun 2007 16:59:52 GMT
View Forum Message <> Reply to Message

Good job

Mirek

Subject: Re: Threading example for U++
Posted by arturbac on Sat, 30 Jun 2007 10:48:41 GMT

I bit extened:
- More factal sets
- Logarythmic color scale
- one rendering method for all fractals
- fractal calculating methods are callbacks for thread rendering methods
- left down + mouse move , moves the fractal
- left down + CTRL + mouse move selects area for zoom
- axis describing complex area

File Attachments

1) Threadsv4_bin.7z, downloaded 676 times
2) Threadsv4_src.7z, downloaded 665 times

Subject: Re: Threading example for U++
Posted by mirek on Sat, 30 Jun 2007 15:59:56 GMT

Starts to look really nice. It is not an example I am searching for 'reference', but it is now very nice demonstration example...

BTW, Rect_<double> is 'Rectf'. Rect_, Point_, Size_ were introduced only as implementation technique for Point, Point16, Pointf etc...

Subject: Re: Threading example for U++
Posted by mirek on Sat, 30 Jun 2007 16:09:59 GMT

BUG: When moving the area, if you move the mouse outside the view, it gets "stuck" and continues moving even if mouse button is released...

Instead of 'moving' flag, use SetCapture and IsCapture.

You might also try to make the program structure a little bit less complex, I have noticed the troubles you have with routing mouse messages.

One option is to "reverse" class-member and widget-parent-child relations. In U++, you can easily contain the main window IN the child ctrl class:

```
struct Threads : Ctrl {
    TopWindow win;
```

```
    void Paint(...);
    void LeftDown(...);

    Threads() { win.Add(*this); ... }
};
```

This would significantly reduce the number of line in the example (I think).

Subject: Re: Threading example for U++
Posted by arturbac on Sat, 30 Jun 2007 16:26:48 GMT
View Forum Message <> Reply to Message

[quote title=luzr wrote on Sat, 30 June 2007 18:09]BUG: When moving the area, if you move the mouse outside the view, it gets "stuck" and continues moving even if mouse button is released...

Instead of 'moving' flag, use SetCapture and IsCapture.
[/quote
I check this out.

[quote]
You might also try to make the program structure a little bit less complex, I have noticed the troubles you have with routing mouse messages.
[/quote]

I don't have troubles, i jut follow familiar for me .NET rules of connecting events, and similar to Qt.
Control got events which can be connected, thats all.
I didn't use such techniques of letting sub control owning entire window. It don't looks good for me.
Control owns window, window contains control which contains this window. Circular dependency ?
What in case when window got multiple controls and we want to connect many events ?

Subject: Re: Threading example for U++
Posted by mirek on Sat, 30 Jun 2007 18:17:57 GMT
View Forum Message <> Reply to Message

Quote:
Quote:
You might also try to make the program structure a little bit less complex, I have noticed the troubles you have with routing mouse messages.

I don't have troubles, i jut follow familiar for me .NET rules of connecting events, and similar to Qt. Control got events which can be connected, thats all.
I didn't use such techniques of letting sub control owning entire window. It don't looks good for me.

That is just mind barrier, believe me. The key to understand is that class containment does not have to be related to the GUI containment.

Quote:
Control owns window, window contains control which contains this window. Circular dependency ?
What in case when window got multiple controls and we want to connect many events ?

I think "got" is a wrong word here.... Sure, nothing prevents you to follow GUI in class composition, but in U++ it is not requirement.

Before going deep into philosophic details, "inverse layout" in your example would very likely produce much less complex code - which is the thing I am usually pursuing...

Subject: Re: Threading example for U++
Posted by mirek on Sat, 30 Jun 2007 22:31:50 GMT
View Forum Message <> Reply to Message

P.S.:

String FormatTS (int ms) { return Format("Render time %d:%02.2d.%03d",(ms/1000/60), (ms / 1000) % 60, ms % 1000);}

Subject: Re: Threading example for U++
Posted by arturbac on Sun, 01 Jul 2007 18:40:41 GMT
View Forum Message <> Reply to Message

luzr wrote on Sun, 01 July 2007 00:31P.S.:

String FormatTS (int ms) { return Format("Render time %d:%02.2d.%03d",(ms/1000/60), (ms / 1000) % 60, ms % 1000);}

What's wrong with this ?

Subject: Re: Threading example for U++
Posted by mirek on Sun, 01 Jul 2007 19:15:24 GMT
View Forum Message <> Reply to Message

Well, the original code showed one minute, 3 seconds and 50 miliseconds a 1:3:50 (if I remember well), while I would like to see 1:03.050.

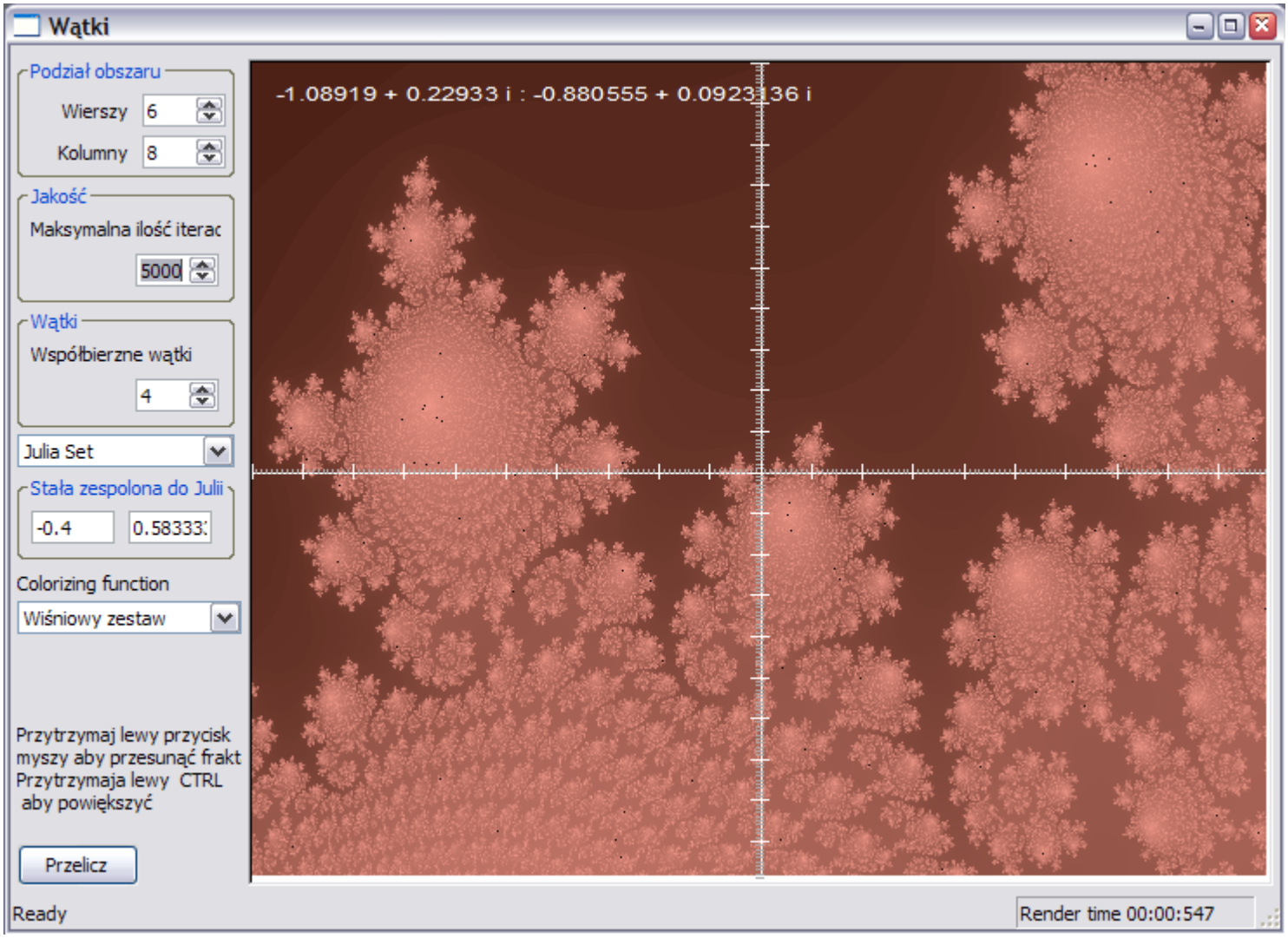
Subject: Re: Threading example for U++
Posted by arturbac on Sun, 01 Jul 2007 21:53:25 GMT
View Forum Message <> Reply to Message

Ok.
Btw the Format example is not finished, i don't known how to format double value with strict number of decimal places.

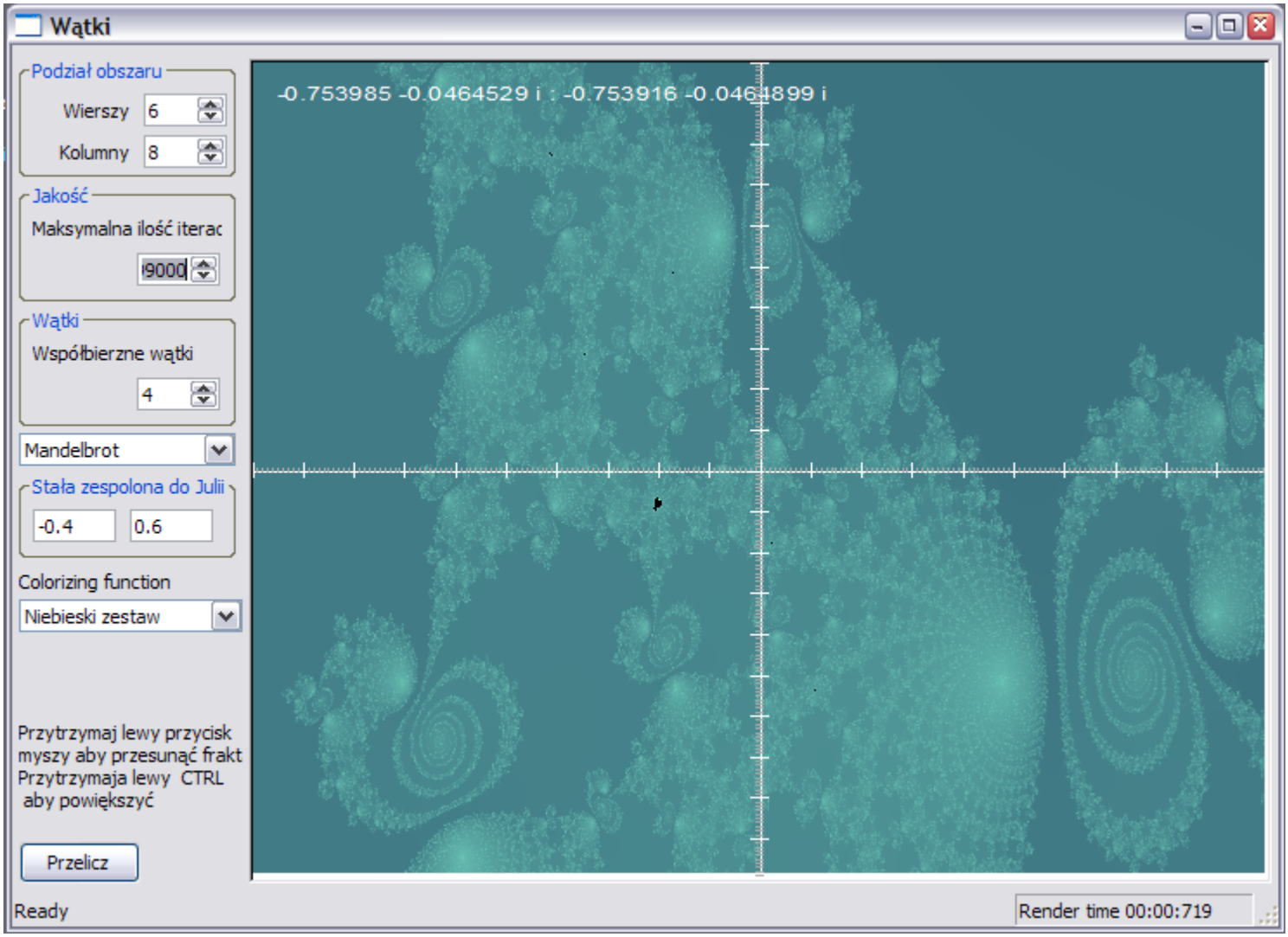
I will next days improve a bit it for ex. many colorize sets etc.

like below.

File Attachments
1) julia5.png, downloaded 2054 times



2) mandel5.PNG, downloaded 2107 times



Subject: Re: Threading example for U++
Posted by mirek on Mon, 02 Jul 2007 06:27:11 GMT
View Forum Message <> Reply to Message

arturbac wrote on Sun, 01 July 2007 17:53Ok.
Btw the Format example is not finished, i don't known how to format double value with strict number of decimal places.

http://www.ultimatepp.org/src$Core$Format$en-us.html

Quote:
printf formatters

Most printf formatters are supported:

c d i o x X ld li lo lx lX e E f g G s

Please refer to printf documentation for the description.

Some more *user* notes about the example:

- option to remove the ruler
- mouse click / Ctrl should be IMO swapped (move with Ctrl, select without)
- 'go back' button

Mirek

Subject: Re: Threading example for U++
Posted by arturbac on Sat, 07 Jul 2007 23:12:48 GMT
View Forum Message <> Reply to Message

A bit extended
SSE2/x86 code for Mandelbrot & Julia

in readme.txt are performance of Code/Compiler test results very interesting, for BSD and Win

- new colorizing functions
- button - switch betwen zoom/move mode

etc.

(things Mirek write I will corect in future)

Subject: Re: Threading example for U++
Posted by nixnixnix on Tue, 17 Jul 2007 01:40:45 GMT
View Forum Message <> Reply to Message

should this code compile on a default install of 2007.1 under windows?

I get

----- plugin\z (GUI MT GCC FORCE_SPEED BLITZ WIN32) (7 / 9)
adler32.c
plugin/bmp: 4 file(s) built in (0:01.67), 417 msecs / file, duration = 3500 msecs, parallelization 100%

```
compress.c
crc32.c
deflate.c
gzio.c
infblock.c
infcodes.c
inffast.c
inflate.c
inftrees.c
infutil.c
trees.c
uncompr.c
zutil.c
----- plugin\png ( GUI MT GCC FORCE_SPEED BLITZ WIN32 ) (8 / 9)
pnglib.c
plugin\z: 14 file(s) built in (0:01.26), 90 msecs / file, duration = 3954 msecs, parallelization 100%
pngupp.cpp
pngreg.icpp
----- ThrFractRend ( GUI MT MAIN GCC FORCE_SPEED BLITZ WIN32 ) (9 / 9)
BLITZ: ThrFractRend_design.cpp ThrFractRend_sse2.cpp ThrFractRend.cpp main.cpp
Paintbox.cpp
plugin\png: 3 file(s) built in (0:04.26), 1420 msecs / file, duration = 8781 msecs, parallelization
100%
Core: 36 file(s) built in (0:11.13), 309 msecs / file, duration = 30922 msecs, parallelization 100%
In file included from C:\MyApps\ThrFractRend\ThrFractRend_design.cpp:42,
                 from  C:/upp/out/ThrFractRend/MINGW.Blitz.Force_speed.Gui.Main.Mt\ $blitz.cpp:3:
C:\MyApps\ThrFractRend\/ThrFractRend.h:84: error: `Mutex' does not name a type
C:\MyApps\ThrFractRend\/ThrFractRend.h:85: error: `Mutex' does not name a type
In file included from  C:/upp/out/ThrFractRend/MINGW.Blitz.Force_speed.Gui.Main.Mt\
$blitz.cpp:10:
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp: In member function `void
mtfr::Threads::FnMandelbrot(const mtfr::ComplexD&, unsigned int*) const':
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:51: error: `__m128d' undeclared (first use this
function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:51: error: (Each undeclared identifier is
reported only once for each function it appears in.)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:51: error: expected `;' before "C0"
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:54: error: `C4' undeclared (first use this
function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:54: error: `_mm_set1_pd' undeclared (first use
this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:55: error: `C0' undeclared (first use this
function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:55: error: `_mm_setr_pd' undeclared (first use
this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:57: error: `X0' undeclared (first use this
function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:60: error: `X1' undeclared (first use this
```

function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:60: error: `_mm_mul_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:61: error: `X2' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:61: error: `_mm_unpackhi_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:62: error: `X3' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:62: error: `_mm_add_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:63: error: `_mm_comigt_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:66: error: `X4' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:69: error: `_mm_sub_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:71: error: `_mm_mul_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:75: error: `_mm_unpacklo_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:76: error: `_mm_add_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp: In member function `void mtfr::Threads::FnJulia(const mtfr::ComplexD&, unsigned int*) const':
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:85: error: `__m128d' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:85: error: expected `;' before "C0"
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:88: error: `C4' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:88: error: `_mm_set1_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:89: error: `C0' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:89: error: `_mm_setr_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:90: error: `X0' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:94: error: `X1' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:94: error: `_mm_mul_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:95: error: `X2' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:95: error: `_mm_unpackhi_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:96: error: `X3' undeclared (first use this function)

C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:96: error: `_mm_add_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:97: error: `_mm_comigt_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:100: error: `X4' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:103: error: `_mm_sub_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:105: error: `_mm_mul_sd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:109: error: `_mm_unpacklo_pd' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend_sse2.cpp:110: error: `_mm_add_pd' undeclared (first use this function)
In file included from C:/upp/out/ThrFractRend/MINGW.Blitz.Force_speed.Gui.Main.Mt\$blitz.cpp:14:
C:\MyApps\ThrFractRend\ThrFractRend.cpp: In member function `void mtfr::Threads::picbox_Paint(Upp::Draw&)':
C:\MyApps\ThrFractRend\ThrFractRend.cpp:82: error: `job_lock' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend.cpp:93: error: `image_lock' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend.cpp: In member function `void mtfr::Threads::ThreadRenderJob(int, int, const Upp::Callback2<const mtfr::ComplexD&, unsigned int*>&, int)':
C:\MyApps\ThrFractRend\ThrFractRend.cpp:131: error: `job_lock' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend.cpp:133: error: `image_lock' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend.cpp:135: error: no matching function for call to `Upp::Semaphore::Release()'
C:/upp/uppsrc/Core/Thread.h:70: note: candidates are: void Upp::Semaphore::Release(int)
C:\MyApps\ThrFractRend\ThrFractRend.cpp: In member function `void mtfr::Threads::Process(const Upp::Callback2<const mtfr::ComplexD&, unsigned int*>&)':
C:\MyApps\ThrFractRend\ThrFractRend.cpp:197: error: `callback4' undeclared (first use this function)
C:\MyApps\ThrFractRend\ThrFractRend.cpp:199: error: `job_lock' undeclared (first use this function)
ThrFractRend: 5 file(s) built in (0:02.48), 496 msecs / file, duration = 4031 msecs, parallelization 16%

There were errors. (1:15.50)

nick

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 17 Jul 2007 09:44:12 GMT
View Forum Message <> Reply to Message

1. 2007.1 is to old i tested it with current u++ in SVN on sourceforge
2. Mingw dosn't have SSE2 intrinsics , (gcc & msvc have) comment out USE_SSE2 in ThreadFractRend.h

Subject: Re: Threading example for U++
Posted by mirek on Tue, 17 Jul 2007 12:52:56 GMT
View Forum Message <> Reply to Message

arturbac wrote on Tue, 17 July 2007 05:441. 2007.1 is to old i tested it with current u++ in SVN on sourceforge
2. Mingw dosn't have SSE2 intrinsics , (gcc & msvc have) comment out USE_SSE2 in ThreadFractRend.h

You can add #if defined(COMPILER_GCC) && defined(PLATFORM_WIN32) too..

Subject: Re: Threading example for U++
Posted by arturbac on Tue, 17 Jul 2007 22:28:40 GMT
View Forum Message <> Reply to Message

Latest Code can be downloaded from this page

http://www.ebasoft.com.pl/index.php?option=com_content&t ask=view&id=21&Itemid=2

Subject: Re: Threading example for U++
Posted by bonami on Wed, 08 Aug 2007 00:59:27 GMT
View Forum Message <> Reply to Message

this is useful, though it does not show everything i need about thread, like terminating one and communiations b/t threads.
thans anyway

Subject: Re: Threading example for U++

Posted by arturbac on Wed, 08 Aug 2007 21:38:59 GMT
View Forum Message <> Reply to Message

1.Terminating - wrong idea (becouse of allocated resources by threads)
Telling thread to stop via some variable - is good idea
2. Comunicating betwen threads for example with ThreadSafeQueue
when one thread do somting for other thread

Will think about changing the example in the future for use with ThreadSafeQueue...

Subject: Re: Threading example for U++
Posted by bonami on Thu, 09 Aug 2007 03:35:58 GMT
View Forum Message <> Reply to Message

arturbac wrote on Thu, 09 August 2007 05:381.Terminating - wrong idea (becouse of allocated resources by threads)
Telling thread to stop via some variable - is good idea
2. Comunicating betwen threads for example with ThreadSafeQueue
when one thread do somting for other thread

Will think about changing the example in the future for use with ThreadSafeQueue...

where is ThreadSafeQueue? not in U++ or your example?
it seems i can use select(). but i still don't know how to put this kind of queue or sth. together with a SOCKET into fd_set

Subject: Re: Threading example for U++
Posted by arturbac on Thu, 09 Aug 2007 19:59:36 GMT
View Forum Message <> Reply to Message

I will put here my fresh one hoever this [C++] implementation is not tested, but Same Generic in C# works well.
[Im at porting stage of some app to c++ so why it is stil not tested]

```
template <typename T>
class ThreadSafeQueue
{
 class QueueElement;
public :
```

```
 int       GetCount() const;
 void      Enqueue(T value);
 T       Dequeue();
 void      FinishWaiting(bool end_job);

 ThreadSafeQueue();
 ~ThreadSafeQueue();

private:
 int       m_count;
 QueueElement *    m_head;
 QueueElement *    m_tail;
 Threading::Monitor   m_monitor;
 Threading::CriticalSection m_lock;
 bool      m_finish_wait;


 class QueueElement
 {
 private:
  friend class    ThreadSafeQueue;

  T       m_value;
  QueueElement *    m_next;

  T       GetValue() const;

  QueueElement(const T & value);
 };
};




template <typename T>
int ThreadSafeQueue<T>::GetCount() const
{
 return this->m_count;
}

template <typename T>
ThreadSafeQueue<T>::ThreadSafeQueue()
 : m_head(NULL)
 , m_finish_wait(false)
 , m_tail(NULL)
 , m_count(0)
{}
template <typename T>
```

```
ThreadSafeQueue<T>::~ThreadSafeQueue()
{
 m_lock.Enter();
 //Clenup remaining items
 if (this->m_head != NULL)
 {
  QueueElement *p = this->m_head, *pdel;
  while (p != NULL)
  {
   pdel = p;
   p = p->m_next;
   delete pdel;
  }
  this->m_head = NULL;
  this->m_tail = NULL;
 }
 m_lock.Leave();
}
template <typename T>
void ThreadSafeQueue<T>::Enqueue(T value)
{
 m_lock.Enter();
 {
  m_count++;
  QueueElement *enqueue = new QueueElement(value);
  if (m_head == NULL)
  {
   m_head = enqueue;
   m_monitor.Pulse();
  }
  else
   m_tail->m_next = enqueue;
  m_tail = enqueue;
 }
 m_lock.Leave();
}
template <typename T>
T ThreadSafeQueue<T>::Dequeue()
{
 QueueElement *dequeued;
 m_lock.Enter();
 {
  while (m_head == NULL && !m_finish_wait)
   m_monitor.Wait();
  if (m_head == NULL && m_finish_wait)
  {
   m_lock.Leave();
   return T(); //End
```

```
  }
  m_count--;
  dequeued = m_head;
  m_head = m_head->m_next;
 }
 m_lock.Leave();
 T res(dequeued->GetValue());
 delete dequeued;
 return res;
}
template <typename T>
void ThreadSafeQueue<T>::FinishWaiting(bool end_job)
{
 m_lock.Enter();
 {
  m_finish_wait = end_job;
  m_monitor.Pulse();
 }
 m_lock.Leave();
}

template <typename T>
ThreadSafeQueue<T>::QueueElement::QueueElement(const T & value)
 : m_next(NULL)
 , m_value(value)
{}

template <typename T>
T ThreadSafeQueue<T>::QueueElement::GetValue() const
{
 return this->m_value;
}



class Monitor : public CriticalSection
{
public:

 Monitor();

 void      Wait() throw(...);
 void      Pulse() throw(...);

private:
 int       m_QueueSize;
 CriticalSection    m_lock;
 Semaphore     m_waiting;
```

```
};

Monitor::Monitor() : m_QueueSize(0) {}
void Monitor::Wait()
{
 m_lock.Enter();
 m_QueueSize++;
 m_lock.Leave();
 m_waiting.Wait();
}
void Monitor::Pulse()
{
 m_lock.Enter();
 if (m_QueueSize > 0)
  m_waiting.Release();
 m_lock.Leave();
}
```

Subject: Re: Threading example for U++
Posted by bonami on Fri, 10 Aug 2007 06:46:10 GMT
View Forum Message <> Reply to Message

i seriously need some advice how to implement my project.
a UDP client. interface is a thread i suppose. & i'd like my client to run in a thread. thus, i donno how to communicate b/t them. in my thread, i need to watch both the network socket and something between UI. if linux, i can select(), but in U++, what? i plan to use a std::queue which UI writes and my thread reads and this still seems not so fit into U++.
any suggestions? thank you very much

Subject: Re: Threading example for U++
Posted by mirek on Fri, 17 Aug 2007 17:45:26 GMT
View Forum Message <> Reply to Message

bonami wrote on Fri, 10 August 2007 02:46i seriously need some advice how to implement my project.
a UDP client. interface is a thread i suppose. & i'd like my client to run in a thread. thus, i donno how to communicate b/t them. in my thread, i need to watch both the network socket and something between UI. if linux, i can select(), but in U++, what? i plan to use a std::queue which UI writes and my thread reads and this still seems not so fit into U++.
any suggestions? thank you very much

Why do not you just use "select" ?

Mirek

Subject: Re: Threading example for U++
Posted by bonami on Tue, 04 Sep 2007 07:05:49 GMT
View Forum Message <> Reply to Message

i prefer Win32 programming. so i suppose WaitForSingleObject or sth. should be what I need. since i never win32 programmed before, i have trouble here. if i both WSAStartup and select(), i suppose it is a mix of styles and thus puzzling.

Anyway, as a last resort, select() is acceptable for me, too. however in this case, i don't know how to inter-thread communicate using select() since i need to communicate using SOCKET