

---

Subject: How to close Skylark server except using Ctrl+C

Posted by [kasome](#) on Tue, 17 Jul 2012 02:46:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

i am trying to use the Skylark to do something like the following

```
class WebServer : public SkylarkApp {
public:
    void StopServer();
    void StartServer();

};

class SimpleGUI : public TopWindow , public WebServer {
public:
    SimpleGUI() {
        StartServer();
    }

    virtual ~SimpleGUI() {
        StopServer();
    }
}
```

GUI operation for desktop user, and web operation for smart phone user.

so i try to modify the code, like

```
class WebServer : public SkylarkApp {
public:
    WebServer() {
        Run();
    }
    virtual ~WebServer(){
    }

    void StopServer() {
        TheApp().quit = true;

        TcpSocket h;
        h.Connect("127.0.0.1", TheApp().port);
        h.Put("quit");
    }
}
```

```
}  
void StartServer() {  
    for( int i=0; i<threads; ++i ) {  
        Thread::Start( THISBACK(ThreadRun) );  
    }  
}  
  
virtual void Main() {  
  
}  
  
};
```

and it didn't work apparently,

well, is there better way to close Skylark server except using Ctrl+C ?

---

---

Subject: Re: How to close Skylark server except using Ctrl+C  
Posted by [mirek](#) on Sun, 02 Sep 2012 08:14:08 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, at this moment, this usage scenario is unexplored territory.

Anyway, your method does not look that bad, what are symptoms?

Mirek

---

---

Subject: Re: How to close Skylark server except using Ctrl+C  
Posted by [Zbych](#) on Sun, 05 May 2013 18:20:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

I made some tests and it is possible to close skylark nicely, but there are two problems. First - quit flag is private, second "main" function waits for all threads in application.

Mirek, can you add Quit (public) method to skylark and modify Main?

```
void SkylarkApp::Quit()  
{  
    quit = true;  
#ifdef PLATFORM_POSIX  
    Broadcast(SIGTERM);  
}
```

```

#endif
TcpSocket s;
s.Timeout(100);
s.Connect("127.0.0.1", port);
}

void SkylarkApp::Main()
{
Buffer<Thread> uwt(threads);
for(int i = 0; i < threads; i++)
    uwt[i].Run(THISBACK(ThreadRun));

/* Wait for threads to shut down */
for (int i = 0; i < threads; i++)
    uwt[i].Wait();
}

```

---

Subject: Re: How to close Skylark server except using Ctrl+C  
 Posted by [nneilson](#) on Tue, 07 May 2013 02:30:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

At least for my application in C++ and Java is to shut down threads INSIDE each individual thread.

```

void endX(){
    endApp = true;
    snd("clear61", 0);
    Sleep(200);
    snd("clear62", 0);
    Sleep(200);
    snd("clear63", 0);
    Sleep(200);
    snd("clear64", 0);
    Sleep(200);
    snd("clear", 0);
    Sleep(2000);
    if(CommPort1.IsOpened()) CommPort1.Close();
    if(CommPort2.IsOpened()) CommPort2.Close();
}

GUI_APP_MAIN{
    NefbG2().Run();
    endX();
}

```

```
}
```

And in each thread something like this:

```
while(!inF.IsEof()){  
    if(endApp){if(inF) inF.Close(); break;}  
    if(Thread::IsShutdownThreads() || cF==1){  
        if(inF) inF.Close();  
        cF=0;  
        break;  
    }  
}
```

Hope this helps as if an app opens a port and it is not closed properly it stays open when the app closes, plus other things happen that IS NOT nice to contend to.

edit: As mentioned try and close the threads inside the thread.

This is when the app exits or even to end a thread while the app is running.

---

---

Subject: Re: How to close Skylark server except using Ctrl+C

Posted by [mirek](#) on Tue, 07 May 2013 07:45:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Thanks, patch applied.

Mirek

---