INTEGRATION FORTH

STEP ONE

Eiveo engine provide possibility to choose of scripting languages. I could not miss this opportunity. I hope and I plan add Forth programming language as additional script language in the Eiveo, then rewrite JS-scripts in Forth and finally avoid JS-interpreter at all. Hard issue. It requires knowledge and expirience. Necessary be studied material and practice. In order to self-education this is first steps, first expirience of integration of Java-application with Forth translator. Certainly result is not very impressive. But it is just first step.

As Forth-system I take Quark-Forth yet. This choise is due to only ease integration. It's just works without troubles. But there is only Windows implementation of it. So this is not portable and is not fittest for practical use, is just for expirements, for self-education.

Quark live in quark.dll. Java native interface require certain kind of function names in the dinamic link libraries look like : “Java_ClassName_FunctionName”. This is not possible to inplement in the quark.dll because it is closed code. So I wrote Qwcdll.dll - wrapper around the quark.dll which export the correct names. Also Qwcdll.dll provides the console input/output via regular Windows console instead of default OpenGL Quark console which is not quite usable yet unfortunately.

There is primitive Java-application FUser.java. FUser contain one “API” function api_proc_1(), which can be called from Forth.

How to it work :

1. First, FUser download dinamic library Qwcdll.dll which is wrapper around the Forth-system. Qwcdll.dll exports Java_Qwcdll_qwc() function.

2. Then FUser call exported function Java_Qwcdll_qwc() in Qwcdll.dll.

3. Java_Qwcdll_qwc() download dinamyc library quark.dll which contains the Forth-system, and then get address of Qwark API procedures (like Evaluate()).

4. Then Java_Qwcdll_qwc() put pointer of qwc_emit() function to top of Quark data stack and Quark reload hiself EMIT vector word as call of this function. qwc_emit() it is alternative implementation of the EMIT semantic which out char to regular Windows console instead of default Quark console. This provide redirection almost all Quark output to Win console.

5. Then Java_Qwcdll_qwc() put pointer of gameengine_api_proc_1() function to top of Quark data stack and Quark define ENG-API-PROC-1 word as call of this function. gameengine_api_proc_1() live in Qwcdll.dll. gameengine_api_proc_1() (in Qwcdll.dll) just call api_proc_1() in Java application. Theoreticaly we can call api_proc_1() directly whithout intermediate gameengine_api_proc_1() but I think this has more disadvantages than advantages.

6. Then Java_Qwcdll_qwc() provide console input/output in circle via standard C function gets() and Quark's Evaluate().

See details in source codes. It is raw variant. I will refactor it. So soon almost all above will obsolete.

IFOne dir content :

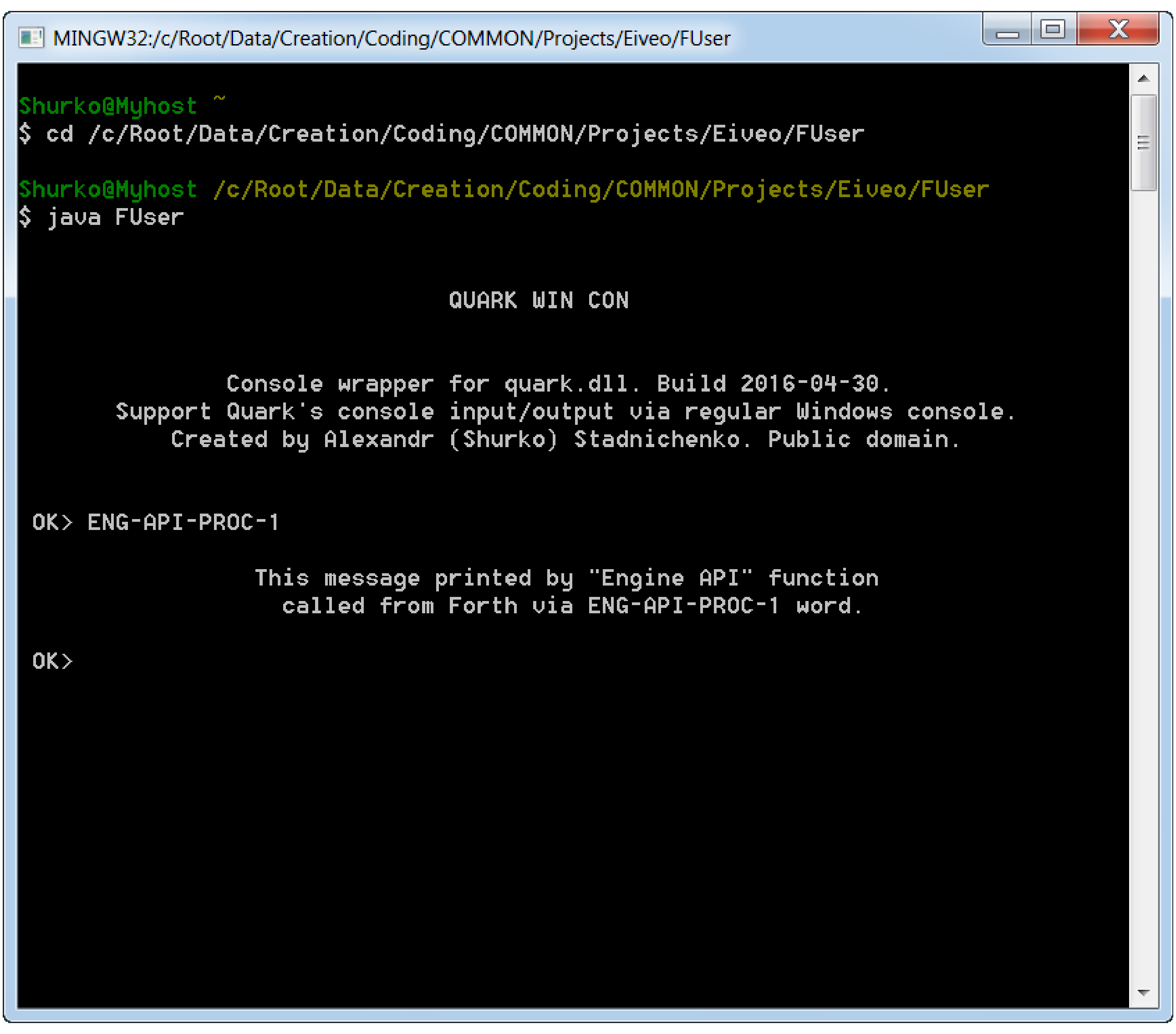
FUsrc		
	FUser.java	Source code.
	Qwcdll.java	Source code.
FUser		
	FUser.class	Compiled code.
	Qwcdll.class	Compiled code.
	Qwcdll.dll	Wrapper in dll.
	quark.dll	Quark system.
	Qwc.exe	Executable Quark wrapper. Just for fun.
	launch.bat	Launcher.
Qwcdll		
	*	Files for creation MSVS project.
	qwcdll.cpp	Source code of wrapper.
Quark-1.0.10.30		
	quark.dll	Quark system.
	quark.exe	Quark with hiself OpenGL console by author.
	quark.pdf	Quark documentation by author (rus).
IFOne.pdf		This document.

How to launch. That something like :

```
C:\> cd Some_Path\FUser
c:\Some_Path\FUser> java FUser
```

Or try **launch.bat** .

Need Java runtime environment.
May need MS VC Redistributable (or not, I don't know) because wrapper written in C++ in MSVS.



MINGW32:/c/Root/Data/Creation/Coding/COMMON/Projects/Eiveo/FUser

```
Shurko@Myhost ~
$ cd /c/Root/Data/Creation/Coding/COMMON/Projects/Eiveo/FUser

Shurko@Myhost /c/Root/Data/Creation/Coding/COMMON/Projects/Eiveo/FUser
$ java FUser


                              QUARK WIN CON


                Console wrapper for quark.dll. Build 2016-04-30.
            Support Quark's console input/output via regular Windows console.
                Created by Alexandr (Shurko) Stadnichenko. Public domain.


 OK> ENG-API-PROC-1

                 This message printed by "Engine API" function
                   called from Forth via ENG-API-PROC-1 word.

 OK>
```

Known bugs :

1. quark.dll's Done( ) function cause crash when called from Qwc. When Done( ) called from SPF 4.2 forth system this not cause crash but have no any effect. Theoretically Done( ) should release the allocated resources.

2. Default BYE and BYE1 words cause crash when called from Qwc. So BYE redefined in Qwcdll.dll to do nothing :

   ```
   '  NOOP  TO  BYE            \  Now BYE vector point to NOOP which do nothing.
   ```

   Closing via Ctrl+C .

   BYE1 called from SPF 4.2 work fine.

In the plans :

1. Migrate to MinGW.
2. Migrate to FICL forth system.

Created with : Oracle Jdk and Jre, JetBrains IntelliJ Idea 14.14 CE, MS Visual C++ 2010 Express.

Alexandr (Shurko) Stadnichenko, 16-05-05.