Subject: Incorrect SHA1 checksum for files 4GB+
Posted by Zbych on Mon, 21 May 2018 12:36:32 GMT
View Forum Message <> Reply to Message

Hi,

Did anyone test SHA1 using files 4GiB or larger?
When I calculate a sha1 for file 4294967295 bytes long, it is the same as returned by sha1sum from linux.
When file is 4294967296 bytes long (or longer) sha1 from Upp and sha1sum differs.

```
#include <Core/Core.h>

using namespace Upp;

CONSOLE_APP_MAIN
{
    constexpr int chunk = 1024 * 1024;
    constexpr int progress_interval = 1000;
//    static const char * filename = "/tmp/file4GB-.bin";
    static const char * filename = "/tmp/file4GB.bin";

    StdLogSetup(LOG_COUT | LOG_TIMESTAMP);
 RLOG("File '" << filename << "' sha1 calculation started");

    Sha1Stream sha1;
    FileIn file(filename);
    auto size = file.GetSize();
    auto last_progress = msecs();

    while (!file.IsError() && !file.IsEof()) {
        auto buff = file.Get(chunk);
        if (buff.GetCount() <= 0) break;
        sha1.Put(buff);
        if (msecs(last_progress) > progress_interval) {
            last_progress = msecs();
            if (size > 0) RLOG("Progress: " << 100 * file.GetPos() / size << "%");
        }
    }

    if (!file.IsError()) RLOG("SHA1: " << sha1.FinishString());
    else RLOG("File '" << filename << "' sha1 calculation interrupted by error: '" <<
file.GetErrorText() << "'");
}
```

Test files can be generated with following commands:

```
openssl rand 1073741824 > /tmp/file4GB-.bin
openssl rand 1073741824 >> /tmp/file4GB-.bin
openssl rand 1073741824 >> /tmp/file4GB-.bin
openssl rand 1073741823 >> /tmp/file4GB-.bin

openssl rand 1073741824 > /tmp/file4GB.bin
openssl rand 1073741824 >> /tmp/file4GB.bin
openssl rand 1073741824 >> /tmp/file4GB.bin
openssl rand 1073741824 >> /tmp/file4GB.bin
```
