
Subject: Using CoFor with SSH package to parallelize sftp file downloads.

Posted by [Oblivion](#) on Sun, 04 Sep 2022 20:34:30 GMT

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

Core/SSH package has some nice multithreading abilities which can be sometimes overlooked..

The code snippet below demonstrates a way to combine the MT capabilities of SSH package with Upp::CoFor (loop parallelization function):

- 1) It reads the content of a remote directory
- 2) Downloads multiple files with certain attributes in parallel, using the CoFor loop.

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

using namespace Upp;

constexpr const char *path = "/pub/example/";
constexpr const int  MAXDOWNLOAD = 4;

CONSOLE_APP_MAIN
{
  StdLogSetup(LOG_COUT|LOG_FILE);
  // Ssh::Trace();

  SshSession session;
  if(!session.Timeout(10000).Compression().Connect("demo:password@test.rebex.net:22")) {
    RLOG(session.GetErrorDesc());
    return;
  }

  SFtp browser(session);
  SFtp::DirList ls;

  // Get a remote dir listing.
  if(!browser.ListDir(path, ls)) {
    RLOG(browser.GetErrorDesc());
    return;
  }

  // Filter the dir list.
  auto files = FilterRange(ls, [](const SFtp::DirEntry& e) { return e.IsFile() && e.GetSize() <=
65536; });

  // Download the files, using worker threads in parallel.
  CoFor(min(files.GetCount(), MAXDOWNLOAD), [&](int i){
    const SFtp::DirEntry& e = files[i];
```

```
String fpath = AppendFileName(path, e.GetName());
RLOG("Downloading " << fpath);
SFtp sftp(session);
String file = sftp.LoadFile(fpath);
if(sftp.IsError())
    RLOG(Format("Worker #%d: %s", sftp.GetId(), sftp.GetErrorDesc()));
else
    RLOG("File " << e.GetName() << " is successfully downloaded.");
});
}
```

Best regards,
Oblivion
