Subject: [solved] Linux LocalProcess::GetExitCode highly divergent from Windows
Posted by cbpporter on Sat, 02 Sep 2017 08:55:11 GMT
View Forum Message <> Reply to Message

Under Linux, LocalPRocess does not give me the exact exit code the child process gave.

Created this sample app duo:
lpchild package:

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

using namespace Upp;

CONSOLE_APP_MAIN
{
 SetExitCode(-42);
 Cout() << "Exited with " << GetExitCode() << "!\n";
}
```

lpparent package:

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

using namespace Upp;

CONSOLE_APP_MAIN
{
 LocalProcess lp("lpchild");
 String t, tt;
 while (lp.Read(t)) {
  if (t.GetCount())
   tt << t;
 }
 Cout() << "Child said:\n";
 Cout() << tt;
 Cout() << "Child exited with: " << lp.GetExitCode() << "\n";
}
```

output:

```
Child said:
Exited with -42!
Child exited with: 214
```

I saw no other API member that could give the real code.

File Attachments

1) MyApps.tar.gz, downloaded 444 times

Subject: Re: Linux LocalProcess::GetExitCode highly divergent from Windows
Posted by cbpporter on Sat, 02 Sep 2017 09:21:41 GMT
View Forum Message <> Reply to Message

OK, looks like this is not an U++ issue but a standard Linux limitation. Return codes are 8 bit. Welcome to 2017!

I guess I need to come up with a standardized API to report error across Linux and Windows, because 0 is valid on Linux, but other positives are error and positives are success on Windows...
