Subject: Re: Upp::CopyStream reports wrong size.
Posted by Oblivion on Mon, 27 Aug 2018 16:58:26 GMT
View Forum Message <> Reply to Message

Edit: On further thinking, I remove the error checking line since the CopyStream returns the bytes actually read.

Quote:what do you think? (it will reach 100% sooner, but in situation where you close progress immediately after finish it will be at 100% for some time).

Well, I think it might not be a big problem if the "dest" stream is on local machine, but for example, in SFtp::SaveFile(), where the destination stream is a remote file system object, the call might block for a long time or indefinitely (or fail/timeout), and this may be confusing for the client. Also, errors on destination stream should better be checked too. Not to mention the existing two CopyStreams have, for some reason, different block sizes (32 and 64 k). So here is my proposal:

```
int64 CopyStream(Stream& dest, Stream& src, int64 count, Gate<int64, int64> progress, int chunk_size)
{
 int block = (int)min<int64>(count, chunk_size);
 Buffer<byte> temp(block);
 int loaded;
 int64 done = 0;
 int64 total = count;
 while(count > 0 && (loaded = src.Get(~temp, (int)min<int64>(count, block))) > 0) {
  count -= loaded;
  done += loaded;
  if(progress(done, total))
   return -1;
  dest.Put(~temp, loaded);
 }
 return done;
}

int64 CopyStream(Stream& dest, Stream& src, int64 count)
{
 return CopyStream(dest, src, count, Null, 65536);
}

int64 CopyStream(Stream& dest, Stream& src, int64 count, Gate<int64, int64> progress)
{
 return CopyStream(dest, src, count, pick(progress), 65536);
}
```

Would this be ok?
In this way I can re-use this code in SFtp too.

Best regards,
Oblivion.