Subject: plugin/zstd and plugin/zstd_legacy
Posted by mirek on Tue, 26 Mar 2019 13:33:30 GMT
View Forum Message <> Reply to Message

This is not likely to affect many U++ users, but...

In the past, we were early adopters of Yan Collet's zstd compression library. We have adopted prerelease version into plugin/zstd. (Frankly, I needed something like that the app I develop...).

Unfortunatly, later the API and the internal format of zstd has changed, which made me stuck with that prerelease versio for about 2 years, undecided how to solve this (and original version worked just fine too...).

Today I have finally resolved the issue: plugin/zstd is now current version 1.3.8. in the "modern" format and to support older files, I have created "plugin/zstd_legacy". This has the same API for old zstd file format, the disctinction is done by namespace Legacy. So if you have both in application, you can use e.g. Upp::Legacy::ZstdDecompressStream to load old file format and Upp::ZstdDecompressStream to load 'modern' format.

Subject: Re: plugin/zstd and plugin/zstd_legacy
Posted by koldo on Wed, 27 Mar 2019 07:48:20 GMT
View Forum Message <> Reply to Message

Thank you Mirek. Nowadays, do you advice to use zstd better than 7z?

Subject: Re: plugin/zstd and plugin/zstd_legacy
Posted by mirek on Wed, 27 Mar 2019 10:52:47 GMT
View Forum Message <> Reply to Message

koldo wrote on Wed, 27 March 2019 08:48Thank you Mirek. Nowadays, do you advice to use zstd better than 7z?

Thats apples / oranges...

zstd is not about improving compression levels, but about doing high speed compression with zlib levels.

7z (LZMA) has better compression, but zstd is much faster.

On the opposite side of spectrum is LZ4 (by the same author), which is significantly faster than zstd (about twice), but the compression is not that great.

Mirek

Subject: Re: plugin/zstd and plugin/zstd_legacy
Posted by koldo on Wed, 27 Mar 2019 15:29:22 GMT
View Forum Message <> Reply to Message

Thank you for the clarification.