**Android bundle 解压速度优化**

LZ4 算法原理：[Lz4压缩算法学习\_markuszhang的博客-CSDN博客\_lz4](https://blog.csdn.net/MarkusZhang/article/details/125134975)

LZ4 源码：[GitHub - lz4/lz4: Extremely Fast Compression algorithm](https://github.com/lz4/lz4)

iOS:[darwin-xnu/lz4.c at 0a798f6738bc1db01281fc08ae024145e84df927 · apple/darwin-xnu](https://github.com/apple/darwin-xnu/blob/0a798f6738bc1db01281fc08ae024145e84df927/osfmk/vm/lz4.c)

LZ4各种语言的版本：[lz4.github.io](https://lz4.github.io/lz4/#interoperable-lz4)

**C** decoder (tiny)：https://github.com/jart/cosmopolitan/blob/467504308a103865c058f9a0ac858cc22e72240e/libc/nexgen32e/lz4cpy.c

**速度指标对比**

脉脉 Android 端 的实践数据，在一加 7Pro 上有 3 倍的提升：

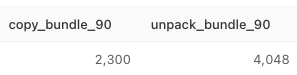


LZ4算法 vs ZIP算法 对比分析：

1、官方 benchmark LZ4 解压速度是 zip 的 10 倍，详见[Japex Report: Test suite for JVM compressors Description Micro-benchmark for testing performance of](https://lz4.github.io/lz4-java/1.8.0/lz4-decompression-benchmark/)

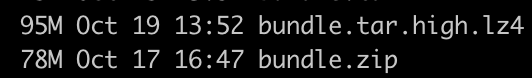
2、没有达到官方 10 倍的原因是脉脉实测对比数据包含了写存储的时间，而写存储的时间zip 和 lz4 是一样的，所以解压+写盘的总体提升只有 3 倍。

3、线上 bundle.zip解压 90 分位时间为 2300+4048=6348ms，数据如下：



**包体积指标对比**

lz4包比 zip 包体积增大 17MB，即增加 22%



**探索 1：LZ4\_Frame\_Format vs LZ4\_Block\_Format**

[lz4/lz4\_Frame\_format.md at dev · lz4/lz4](https://github.com/lz4/lz4/blob/dev/doc/lz4_Frame_format.md)

[lz4/lz4\_Block\_format.md at dev · lz4/lz4](https://github.com/lz4/lz4/blob/dev/doc/lz4_Block_format.md)

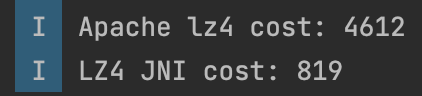
Frame 格式：属于自包含格式，解压无需额外的元数据。

Block 格式：属于非自包含格式，解压需要额外的元数据，比如压缩前的大小，参见[LZ4#metadata](https://github.com/lz4/lz4/blob/dev/doc/lz4_Block_format.md#metadata)。Block 格式的优势是速度比 Frame 更快。

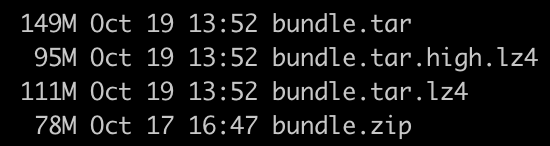
LZ4 官方提供的命令行工具只支持 frame 格式，详见[LZ4#manual](https://github.com/lz4/lz4/blob/86e1ad67528af715ff44ba9954ab250a9898b141/doc/lz4_manual.html)

**探索 2：lz4-java vs apache.commons.compress**

org.apache.commons.compress.compressors.lz4性能很差，比net.jpountz.lz4慢 6 倍！如图所示：



**探索 3：LZ4 压缩参数的选择**



1、原始数据为bundle.tar，149MB

2、zip 包为 78MB

3、bundle.tar.lz4 为 lz4 -1（最低压缩率） 的压缩产物，111MB

4、bundle.tar.hight.lz4 为 lz4 -9 的压缩产物，95MB

5、移动端实测 bundle.tar.lz4 和 bundle.tar.hight.lz4 的解压缩时间几乎一致，但是 bundle.tar.hight.lz4的体积有 16MB 的优势

**探索 4：LZ4FastDecompressor vs LZ4SafeDecompressor**

1、LZ4FastDecompressor在解压缩时是已知源字符串长度，可以解压LZ4\_Block\_Format格式

2、LZ4SafeDecompressor在解压缩时是已知压缩字段的长度，可以解压LZ4\_Frame\_Format格式

3、因为 LZ4 命令行工具只能打出 frame 格式的压缩包，所以必须用LZ4SafeDecompressor解压

4、LZ4FastDecompressor用的是lz4.h中的LZ4\_decompress\_fast，该函数已被废弃，详见：[github.com](https://github.com/lz4/lz4/blob/586e9a4c6db004fe6dc298b508aa364e22996bdb/lib/lz4.h#L841)