



Arm Solutions at Lightspeed

# openEuler上通过 UADK加速大数据应用的 最佳实践

Kevin Zhao - Linaro  
openEuler Summit 2024

# Agenda

- 概述
- UADK-Bigdata应用场景
- UADK-Bigdata集成方式



# 概述

# 概述

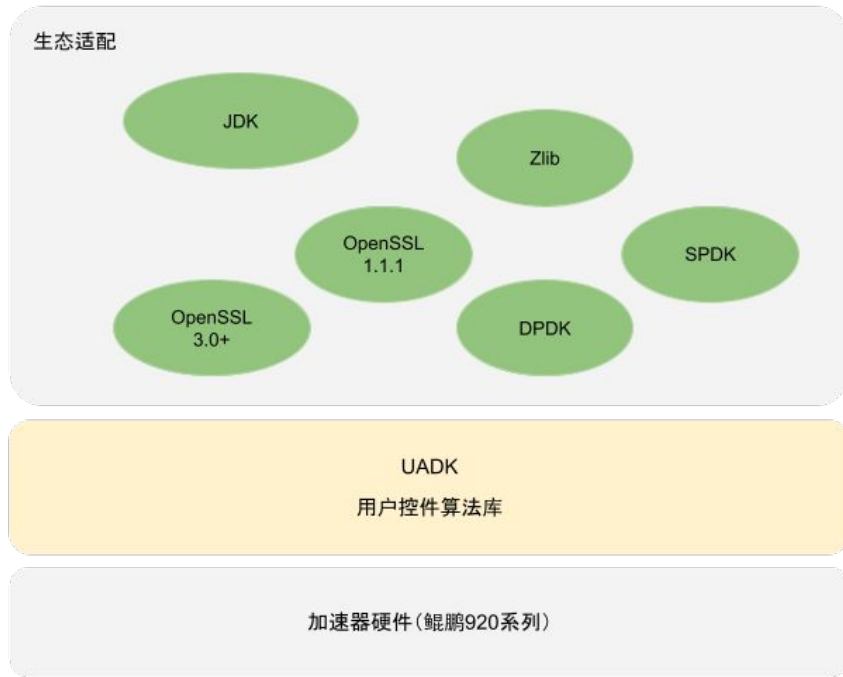
1. 数据规模爆炸式增长
  - a. 数字化转型使全球数据量呈指数级增长, 企业面临存储效率与成本压力
  - b. 数据压缩技术可提升存储效率, 但需要平衡压缩率与处理性能
2. AI时代的数据安全需求
  - a. AI和大数据分析需要海量数据支撑, 数据全生命周期的加密保护成为刚性需求
  - b. 合规要求推动加密需求: 中国的商用密码应用推广, 美国的HIPAA和FISMA规定
3. 硬件加速的价值
  - a. 加解密与压缩都是计算密集型操作, 消耗大量CPU资源
  - b. 专用硬件加速器可显著提升性能, 降低系统负载
  - c. 新型加速架构助力数据安全与AI应用协同发展

# UADK软件生态建设

UADK(User Space Accelerator Development Kit), 支持共享虚拟地址(SVA)技术, 为用户提供高效利用硬件加速器能力的统一编程接口。UADK提供了一组不断扩展的高性能算法实现, 涵盖了加密、压缩等功能。目前已经对接的生态链加速组件:

- OpenSSL 1.1.1f+ /OpenSSL 3.0+
- DPDK/SPDK, 支持UADK crypto PMD 和 UADK compress PMD
- [OpenJDK / BishengJDK](#)
- [Zlib 压缩库](#)
- GmSSL3.0, 服务于国密算法SM2/3/4
- Nginx 1.20.0, 对https短连接场景有很好的加速效果

通过对JDK和zlib库的原生支持, UADK能够更加有利于大数据组件原生应用。





Arm Solutions at Lightspeed

# UADK-Bigdata应用场景

# 大数据领域的应用场景

## 1. 加密

### HDFS Transparent Encryption

在HDFS透明加密中, 首先需要定义[加密区 \(Encryption Zone\)](#), 每个加密区都会使用一个密 钥来加密其中的文件。这些密钥由一个集中的密 钥管理服务(如Apache Ranger或Cloudera Navigator Key Trustee)进行管理, 确保密 钥的安全性和生命周期管理。

当用户访问加密区中的文件时, HDFS透明地对数据进行解密, [用户感受不到加密解密的过程](#)。这样, 即使HDFS的物理存储被非法访问, 数据也因为被加密而保持安全。

在实践中, 我们使用 SM4 作为加密算法。

# 大数据领域的应用场景识别

## 2. 压缩

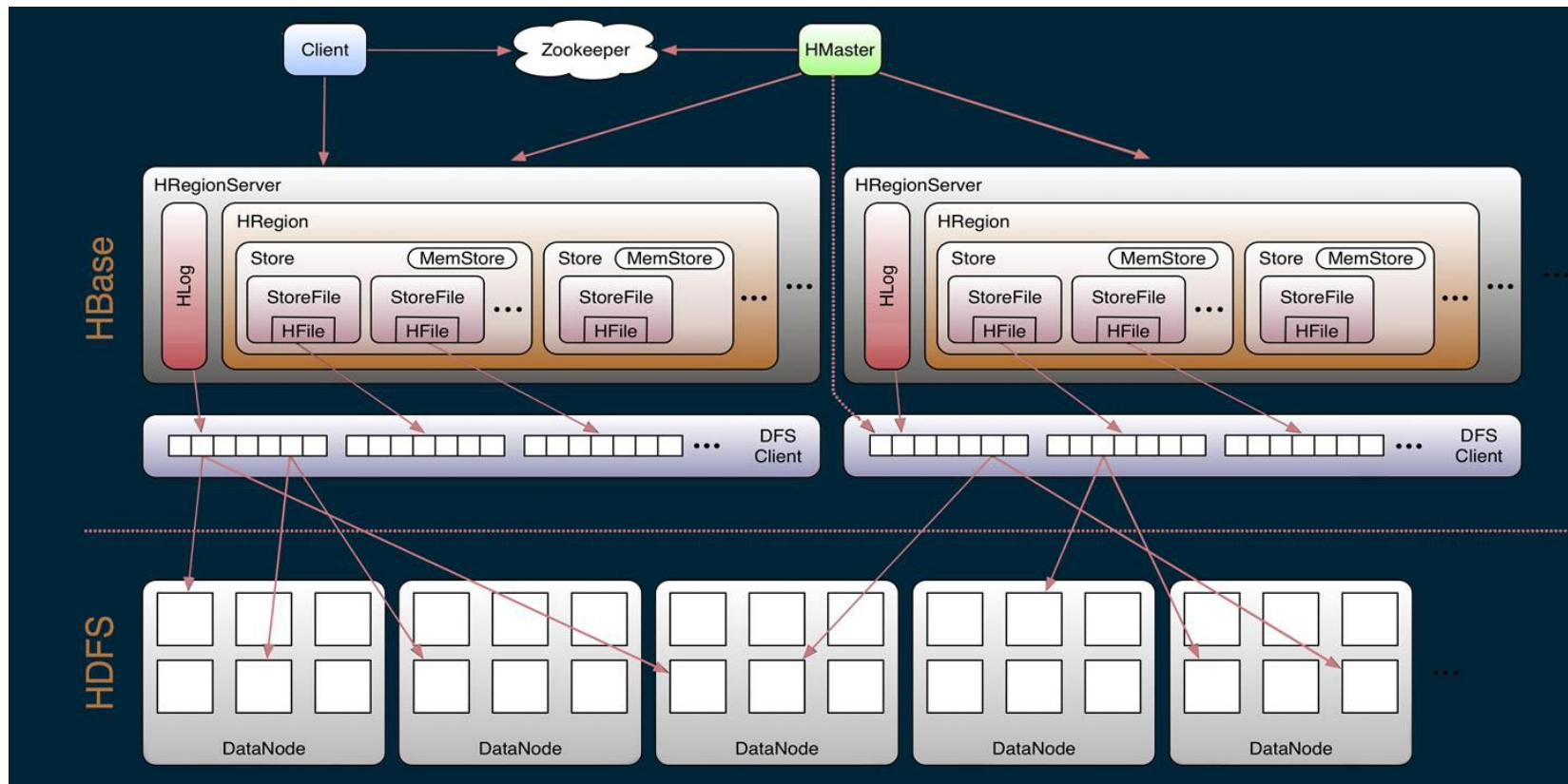
### HBase

作为列式存储数据库，与传统行式数据库比较，通常HBase会占用更大的存储空间，以换取更高的查找和定位速度，满足大数据领域对特定数据检索的要求。

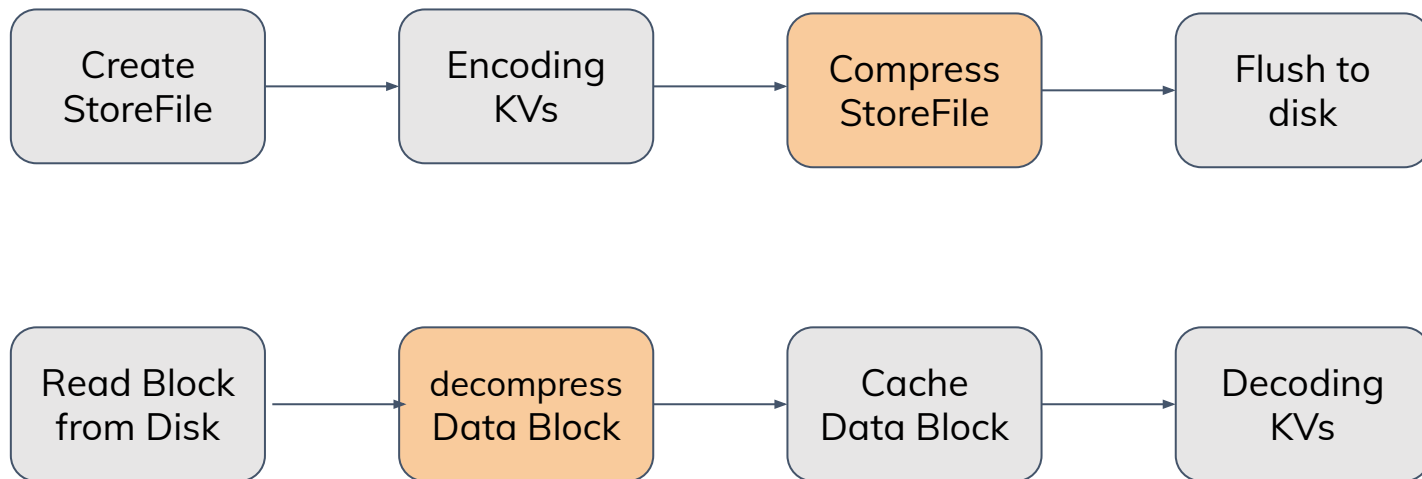
HBase的写入顺序为先写入memstore，再刷入HDFS，即后端存储。为了降低从memstore flush到磁盘的数据大小，在Flush过程中，通常会针对storeFile进行压缩，支持LZ4, SNAPPY, BZIP2, GZIP等压缩算法。



# HBase数据存储结构



# HBase 写入和读取数据过程





Arm Solutions at Lightspeed

# UADK-Bigdata集成方式

# HBase GZ压缩

GZ压缩的配置：

- HBase建表时指定
- HBase PE测试也可以对压缩算法进行测试
- GZ压缩依赖于HadoopNativeLibrary中对Zlib的支持

[zlib-uadk](#):

- Wrap the Zlib to UADK
- 上层应用无感知，无缝兼容zlib生态

# HBase GZ压缩

在Hadoop Native library中加入libz的软链接：

```
[hadoop@hadoop1 native]$ cd /usr/local/hadoop/lib/native/
[hadoop@hadoop1 native]$ ls -la libz.so.1
lrwxrwxrwx. 1 hadoop hadoop 11 Nov 11 16:55 libz.so.1 -> libz.so.1
```

Hadoop checknative 检查native zlib使能状态：

```
[hadoop@hadoop1 native]$ hadoop checknative
2024-11-12 10:41:06,027 INFO bzip2.Bzip2Factory: Successfully loaded & initialized
2024-11-12 10:41:06,033 INFO zlib.ZlibFactory: Successfully loaded & initialized
2024-11-12 10:41:06,039 WARN erasurecode.ErasureCodeNative: ISA-L support is not
2024-11-12 10:41:06,073 INFO nativeio.NativeIO: The native code was built without
Native library checking:
hadoop: true /usr/local/hadoop/lib/native/libhadoop.so.1.0.0
zlib: true /usr/local/hadoop/lib/native/libz.so.1
zstd : true /usr/lib64/libzstd.so.1
bzip2: true /usr/lib64/libbz2.so.1
openssl: false EVP_CIPHER_CTX_block_size
ISA-L: false libhadoop was built without ISA-L support
PMDK: false The native code was built without PMDK support.
```

Hbase native checker检查native zlib使能状态：

```
[hadoop@hadoop1 native]$ hbase org.apache.hadoop.util.NativeLibraryChecker
2024-11-12T10:44:28,655 INFO [main] bzip2.Bzip2Factory: Successfully loaded & initialized native-bzip2 library system-native
2024-11-12T10:44:28,668 INFO [main] zlib.ZlibFactory: Successfully loaded & initialized native-zlib library
```

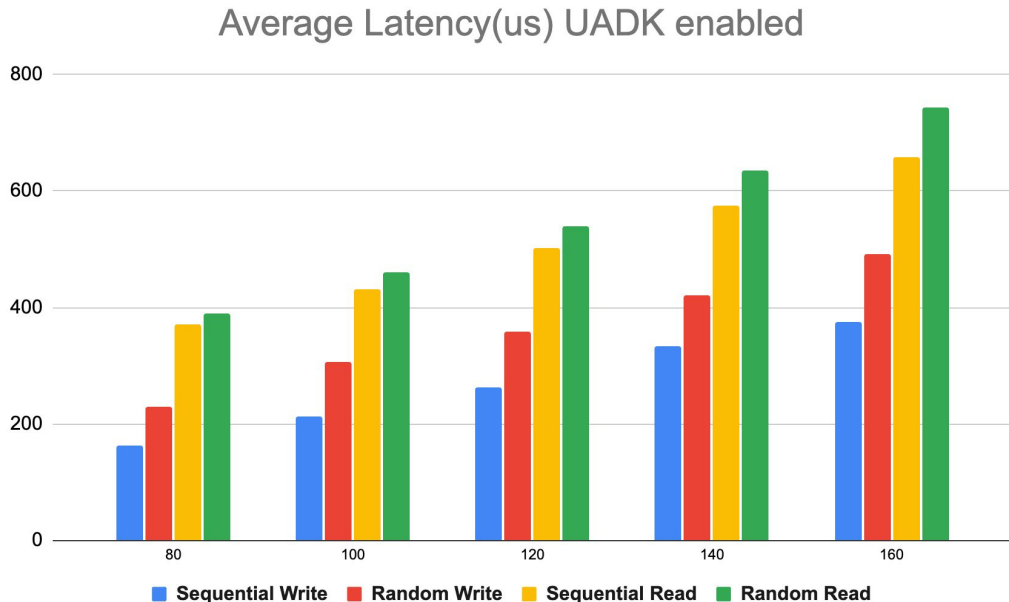
# HBase压缩性能测试

测试集:HBase PE测试

- 随机读写, 顺序读写
- 数据集大小:20G, 50G, 100G
- 线程数:80, 120, 140, 160
- 压缩算法:GZIP
- Hbase/Hadoop伪分布式部署

未使能UADK的情况下, 随机读写, 设置压缩算法为GZIP条件下:

- TPS/QPS大概为5-8w row/s.
- Latency在800-2400 us, 与 threads成正相关。



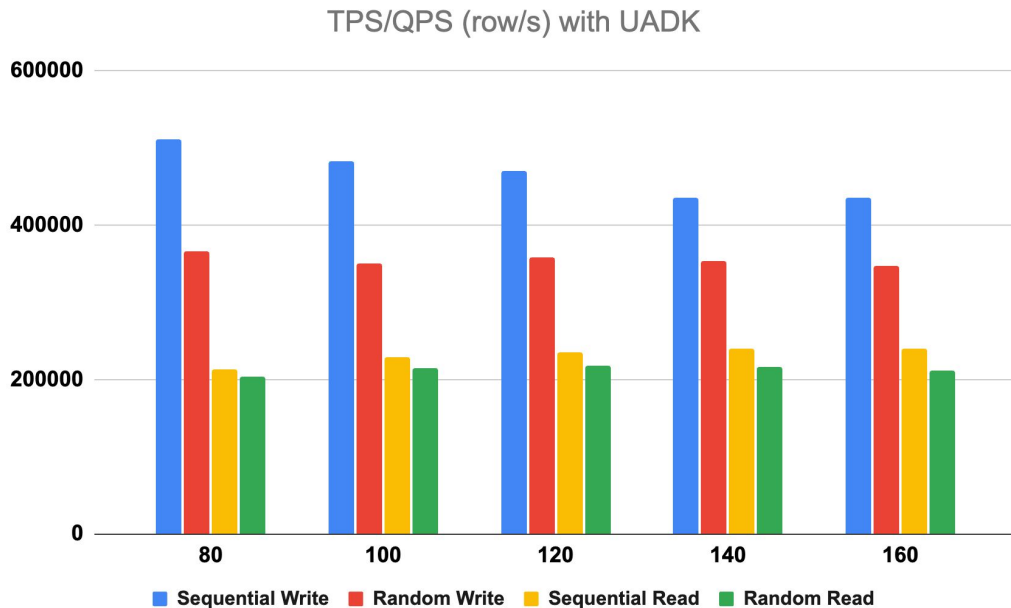
# HBase压缩性能测试

## 测试集:HBase PE测试

- 随机读写, 顺序读写
- 数据集大小:20G, 50G, 100G
- 线程数:80, 120, 140, 160
- 压缩算法:GZIP
- Hbase/Hadoop伪分布式部署

未使能UADK的情况下, 随机读写, 设置压缩算法为GZIP条件下:

- TPS/QPS大概为5-8w row/s.
- Latency在800-2400 us, 与 threads成正相关。

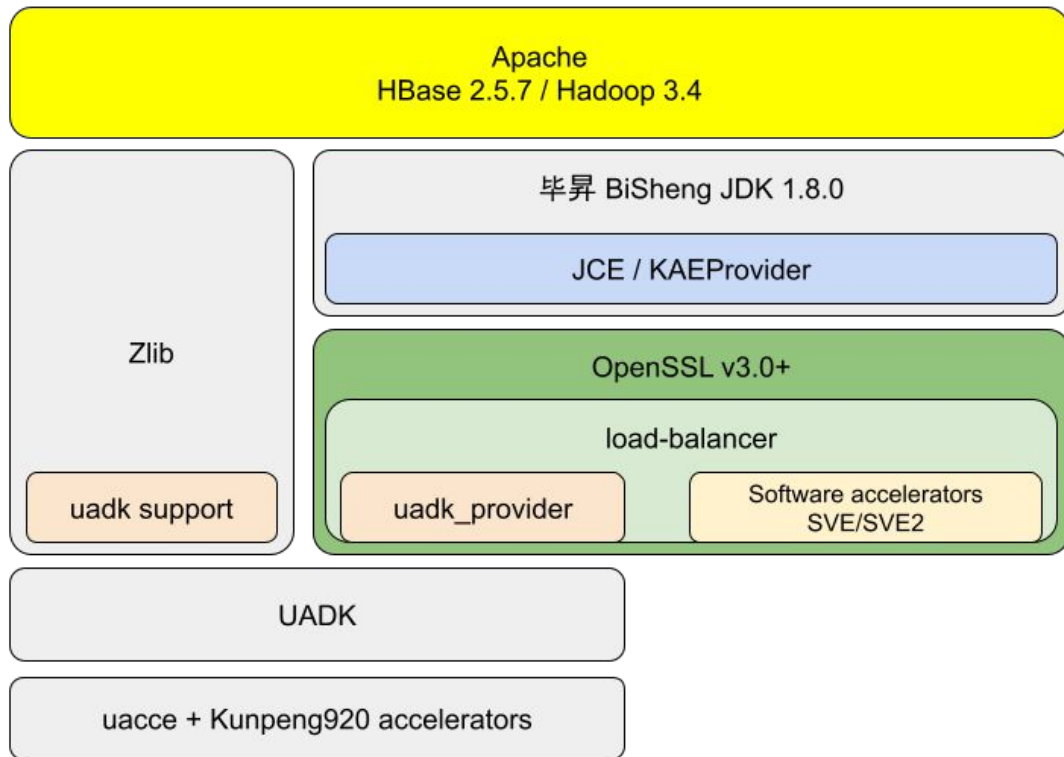


# UADK 与 JDK 的对接 - 加解密

Bring the Abilities to the Java World

⇒ KAEProvider is an encryption interface based on **JCA (Java Cryptography Architecture)** that uses the EVP interface to call the computing power provided by OpenSSL.

⇒ With addition of UADK, it is possible to use hardware accelerations into Java world, powering even more applications.





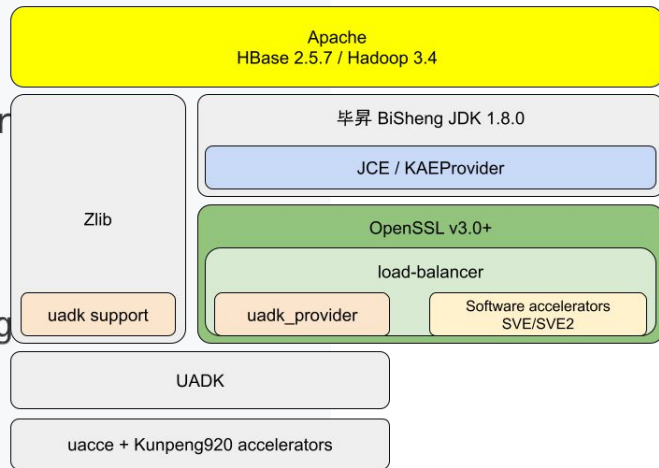
# UADK 与 JDK 的对接 - 加解密

Bring the Abilities to the Java World

**java.security :**

sun.security.provider is an encryption interface based

```
security.provider.1=org.openeuler.security.openssl.KAEProvider
security.provider.2=sun.security.provider.Sun
security.provider.3=sun.security.rsa.SunRsaSign
security.provider.4=sun.security.ec.SunEC
security.provider.5=com.sun.net.ssl.internal.ssl.Provider
security.provider.6=com.sun.crypto.provider.SunJCE
security.provider.7=sun.security.jgss.SunProvider
security.provider.8=com.sun.security.sasl.Provider
security.provider.9=org.jcp.xml.dsig.internal.dom.XMLDSig
security.provider.10=sun.security.smartcardio.SunPCSC
security.provider.11=sun.security.mscapi.SunMSCAPI
```



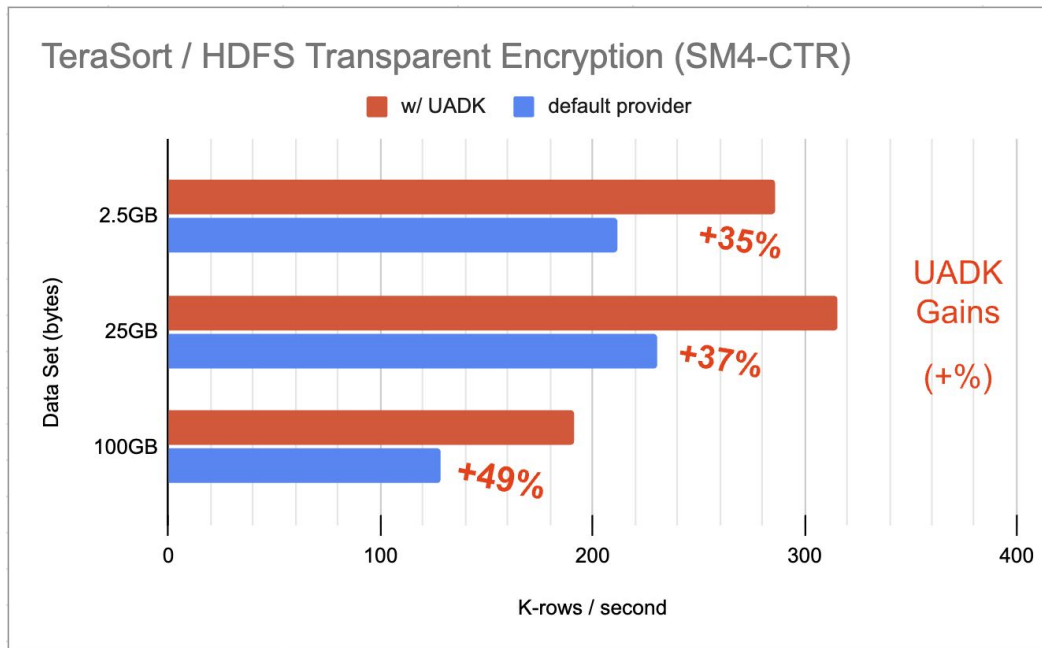
# Benchmark - TeraSort

Measured with 100GB data in TeraSort, on the HDFS Transparent Encryption.

```
# hdfs crypto -getFileEncryptionInfo -path
/zone2/terasort-input/part-m-00000

{cipherSuite: {name: SM4/CTR/NoPadding,
algorithmBlockSize: 16}, ...}

# hadoop jar
$HADOOP_HOME/share/hadoop/mapreduce/hadoop-exam
ples.jar terasort /zone2/terasort-input
/zone2/terasort-output
```



# UADK-BigData 创新项目 - openEuler

BigData + UADK: 加速大数据处理的全栈解决方案

<https://gitee.com/openeuler/uadk-bigdata>



UADK-BigData 创新项目



# Thank you!



Arm Solutions at Lightspeed