mobibench(工作负载发生器)是一个Android应用程序，它生成文件系统以及sqite数据库操作。

mobigen(工作负载重放机)是一种跟踪捕获和重放工具，专门设计用于在不需要实际人工干预的情况下执行用户实验。mobigen记录从用户行为生成的系统调用，并清除可重放表单的跟踪。mobigen可以在不同的Android平台上或使用不同的I/O堆栈配置重放此跟踪。

AndroidI/O栈由DBMS、文件系统、I/O守护进程和I/O调度程序组成。Android平台分别使用sqite、ext 4、mmcqd和CFQ调度器作为DBMS、文件系统、I/O守护进程和I/O调度程序。Android应用程序使用sqite以持久的方式维护信息。Android的默认DBMS sqite在Android平台中生成90%的整个写操作。Jeong等人揭示了AndroidI/o行为的过度低效，这是由ext 4和sqite之间不协调的交互造成的，被称为Journaling of Journal。通过适当的优化，使sqlitdb性能提高了300%。在android中，I/O工作负载的一些典型特征如下：(I)由fsync()跟随的4KB随机写；(II)小文件（小于12KB）的频繁创建与删除。

(fsync函数同步内存中所有已修改的文件数据到储存设备。)

为了刻画Android I/O栈的I/O行为，我们需要正确的工具：(1)正确捕获I/O行为的各种属性：(2)生成代表Android I/O子系统本质行为的综合工作负载；(3)重放I/O跟踪，比较I/O子系统在各种Android设备中的性能。

在为基于Android的设备生成I/O工作负载时，必须考虑四个基本特性。首先，工作负载生成器必须有一个函数来生成write()和fsync()，fsync()是AndroidI/o的一个基本特性。其次，工作负载生成器不仅可以测量文件I/O的性能，而且还可以测量sqite的性能。第三，工作负载生成器必须能够作为Android应用程序以及shell命令运行。(需要使用单独的应用程序来衡量数据库操作(如INSERT、UPDATE或DELETE)的性能。我们提供两个版本的工作负载生成器：作为Android应用程序和shell命令)最后，该工具必须为文件和数据库提供各种选项，以衡量文件I/O和DBMS的性能。

sqite调用fsync()或fdat异步c()以使回滚日志更新和数据库更新持久。

传统的I/O特征研究从四个维度定义了I/O特征：I/O类型(读与写)、I/O大小(Kb)、空间局部性(顺序与随机)和时间局部性(热与冷)。要正确理解Android平台的I/O特性，需要在不同的环境下获得上述四个特征：文件类型、文件系统块类型(如元数据、数据或日志)和Android应用程序(如Facebook、YouTube等)。

1. 确定给定块的文件类型。我们定义了六种文件类型：sqlitdb、sqlitJournal、可执行文件、多媒体文件、资源文件和其他文件。这个过程由两个步骤组成。首先，从逻辑块地址中，提出的工具标识逻辑块地址所属的文件的inode号。然后，从inode编号中识别出相应文件的文件类型；
2. 标识给定块的应用程序。建议的工具可以标识最初发出I/O请求的应用程序。与传统Linux内核不同，AndroidI/o子系统将I/O请求的处理委托给特殊的守护进程mmcqd。从块设备层，mmcqd被视为给定I/O请求的应用程序。我们需要找到给定I/O的原始应用程序，而不是mmcqd。

在实现捕获和重放真实世界工作负载的工具时，最重要的两个因素是重放跟踪的时间准确性和正确处理多线程环境中生成的I/O。

作为Android应用程序，基准测试可以使用由手机供应商预链接的共享sqite库。要将基准测试作为shell命令运行，我们需要静态地将sqite库链接到基准程序。

mobibench提供了详细的基准配置选项，例如分区的选择、线程数和工作负载特性。mobibench在内部存储中使用/data或/sdCard之一，在外部存储卡中使用/ExtSDCard。mobibench允许配置线程数，以提供类似于智能手机的环境，多个线程同时执行I/os和sqite操作。

在mobibench中，可以生成两种类型的工作负载：文件操作和数据库操作。mobibench指定空间局部性(随机和顺序)、I/O模式(读和写)、文件大小、I/O单元大小和同步模式。有五种同步模式：缓冲模式、同步模式、直接模式、mmap模式和write()+fsync()模式。

数据库操作包括插入、更新和删除。这些操作的性能因SQLite的编译和实用化选项而有很大差异。Pragma命令用于设置sqite库的操作模式，如sqite同步模式和日志模式。SQLITE的性能因其期刊模式的不同而有很大差异。Mobibench可用于使用手机制造商提供的库存SQLITE库或预链接SQLITE库。

mobibench生成三个性能值：吞吐量、CPU利用率和上下文交换机的数量。在文件I/O测试中，序列操作的吞吐量为“kb/s”，随机操作为“IOPS”。SQLITE操作的吞吐量单位是“事务/秒”。CPU的使用区分了活动、空闲和io-等待，以了解测试是如何利用CPU的。mobibench还计算上下文切换的数量，以度量上下文切换开销。