

J2ME专业手机游戏开发基础 PDF转换可能丢失图片或格式，  
建议阅读原文

[https://www.100test.com/kao\\_ti2020/238/2021\\_2022\\_J2ME\\_E4\\_B8\\_93\\_E4\\_B8\\_9A\\_c104\\_238968.htm](https://www.100test.com/kao_ti2020/238/2021_2022_J2ME_E4_B8_93_E4_B8_9A_c104_238968.htm) 游戏的基本结构 游戏是什么呢？在游戏中，往往是显示各种各样的画面，玩家可以做一些设定好的控制，画面根据玩家的控制有所变化。从这儿可以看出，游戏至少需要三个功能 - 显示画面，接受玩家输入和对输入产生反馈。这就是常说的渲染，输入输出和逻辑三个模块。将这三个模块组合在一起有很多方法。比如事件驱动 - 画面保持不变直到接受到输入事件，程序进行逻辑运算然后改变画面。而游戏往往不是这样，游戏是时间驱动的。也就是说无论有没有输入，游戏都在不停的循环 - 检查是否有输入，运行逻辑，渲染画面。这里我多说一句，其实什么样的结构并不能区别一个程序是不是游戏，其实只要可以互动娱乐的程序就可以称为游戏，也不一定要用时间驱动，但采用时间驱动是专业游戏的普遍做法，也是很容易接受的方法 - 因为从一定程度上讲，游戏很像电影，随着时间流逝画面在改变。既然是时间驱动，游戏中就会有帧的概念。所谓帧就是某个时刻显示在屏幕上的画面。从整体上看，游戏就是一系列的帧不断播放着，像动画片一样，不过玩家可以通过交互改变播放的内容。而我们开发游戏的主要任务就是安排每一帧的内容。在每一次游戏循环中，我们需要搜集玩家的输入、运行逻辑以更新游戏的数据、根据更新后的数据安排下一帧显示的内容。所以一个最简单的游戏结构就是：  
0 初始化游戏  
1 是否结束游戏 ( Yes:转到6)  
2 搜集玩家输入信息  
3 运行游戏逻辑  
4 更新下一帧，显示下一帧  
5 回到1  
6 清理，结束

游戏这是一个最基本的结构，特别对于比较简单的J2ME游戏来说，这个结构更加有代表性。下面我们将分别讲述专业手机游戏如何实现这个结构中的各个内容。游戏循环的实现我们需要一个进入后就一直循环下去直到游戏结束的结构。线程正好可以实现。最通常的做法是让Canvas实现Runnable接口。于是我们就可以实现run方法。下面是一个run方法简化版

```
public void run(){exitMidlet = false .long startTime = 0 .long  
timeCount = 0 .gameInit() .int curKey = 0 .while (!exitMidlet)  
{startTime = System.currentTimeMillis(). //acquire keyacquireKey()  
 //call game loopgameLoop() //repaint the  
screenrepaint().serviceRepaints().frameCount //lock fps  
timeCount = MIN_DELAY - (System.currentTimeMillis() -  
startTime).timeCount = (timeCount try  
{Thread.sleep(timeCount).} catch (InterruptedException ex)  
{}}endMidlet() .}
```

看到我们的while循环了吗？除非在程序逻辑中设定exitMidlet为true - 那是当玩家选择了退出游戏，我们的游戏将一直运行下去。在while循环之前，gameInit方法的作用是进行游戏初始化 - 比如初始化变量值，载入全局数据，生成全局对象等。在while循环中，我们先是调用了acquireKey方法，这个方法将键盘输入信息进行缓冲以便逻辑中判断按键状态，下面讲会讲到键盘缓冲。gameLoop是我们游戏的主体，每帧中的逻辑运算，图形处理都在这里进行。然后是repaint和serviceRepaints，刷新屏幕 - 新的一帧呈现在屏幕上。最后当跳出while之后，我们执行endMidlet结束这个Midlet。endMidlet的内容只是调用了destroyApp和notifyDestroyed方法。好了整个游戏循环就是这样了，下面

讲分别讲述键盘缓冲和gameLoop如何组织。不过再这之前先让我解释下lock fps。FPS就是Frame per second。为了防止游戏在不同的机器上速度变化太大，我们设定一个最大的FPS值，或者说设置一个每帧至少要花费的时间（这里的MIN\_DELAY）。比如我们设置MIN\_DELAY = 50，那么max FPS =  $1000/50 = 20$  帧/秒。锁定FPS有多种方法，这里的方法是判断如果一帧所有的时间还没达到最大时间，那么就让线程sleep一会儿。顺便在说一下FPS的计算，顾名思义用1000除以一帧所有时间即可，不过要注意的是，一般计算的FPS是平均FPS，所以FPS=累计帧数\*1000/累计花费时间。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)