

让JavaME程序真正RunAnywhere PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022__E8_AE_A9J_avaME_E7_c104_252514.htm "Write once,run anywhere" 是JAVA的口号，但在J2ME平台上做的应用，要想不改动代码就run anywhere，难度是很大的。如果要把一个应用程序做到让大多数的机型都适用，就要考虑到方方面面，其难度是相当大的。比如给百宝箱做游戏，上线机型大多是MIDP1.0的机器，感觉移植中最麻烦的还要数声音部分的API，必须根据各个机型来改动。虽然图象还比较容易做成自适应的，但声音部分就一般就只能根据各个机型来改动。下面提供一种解决方案，可以让J2ME程序在运行时自动加载该机型支持的声音资源并用该机型的声音API来播放。关键问题：1. 各机型提供的播放音乐的API都有所不同，特别是较老的机型。需要在运行时根据机型自动加载。2. 各机型支持的声音的资源文件也不同。需要在运行时根据机型自动加载。3. 各机型的JVM不同，多多少少有一些比较特别的BUG。解决方案：1. 原则：能用标准API就用标准API，不能用的话，就用各个机型自身的API。 // Player types static final int STANDARD = 0. //For MIDI static final int NOKIA = 1. //For ott static final int SAMSUNG = 2. //For mmf static final int NEC = 3. //For MIDI static final String[] supportedPlayerTypes = { "javax.microedition.media.Player", //STANDARD API "com.nokia.mid.sound.Sound", // Nokia "com.samsung.util.AudioClip", //samsung "com.nec.media.AudioClip", //nec }. 下面利用反射机制来动态加

载：`public void determinePlayerType() { // use most -> less use
isSupportSound = true. for (int i = 0. i // try to load a proper sound
Player try { Class.forName(supportedPlayerTypes[i]). //加载当前的
Player类型 playerType = i. //保存加载成功的类的类型 return.
} catch (Exception e) { //加载不成功，说明不支持，继续加载下
一种 e.printStackTrace(). } } isSupportSound = false. }` 2. 下面就可以根据在载成功的类型来加载可以播放的声音资源了 `public
void createPlayer(String name) { if (!isSupportSound) return. switch
(playerType) { case STANDARD: // for MIDI case NEC:
createPlayerFactory("/" name ".mid"). break. case NOKIA: //for ott
createPlayerFactory("/" name ".ott"). break. case SAMSUNG: // for
mmf createPlayerFactory("/" name ".mmf"). break. } }` 3. 对各个机型特有的BUG，是没有什么特别好的办法的，只能各个机型调试。这只能怪厂商了... 该方案优点：在移植的时候就不用改动代码。只要在相应的机型JAR包中保留相关的资源就可以了。这样就不用为了各个机型都折腾一遍了。注意：用 `System.getProperty("microedition.platform")` 来确定机型是不保险的，因为有的机型只是简单地返回J2ME platform。遗留问题：
[1] NecN820 在运行 `Class.forName("javax.microedition.media.Player").` 时候会立刻报“应用程序出错”，而不是抛出“`ClassNotFoundException`”异常。这是该机型JVM的特性（BUG），所以给NecN820的代码中必须注释掉 `javax.microedition.media.Player` 的一切信息。这就得改动代码，有违我们的初衷，的确是个遗憾。（估计NEC的机型都素这样的）
[2] 这个类还有待扩展，以支持更多机型。并加入震动部分的API。理论上可以包含所有的机型

。但实际应用中只要包含需要用到的机型相关API就可以了。
测试机型：在三星E708，MOTOV600，NOKIA 7650
,NecN820（注释掉javax.microedition.media.Player相关内容）上
均测试通过。另：关于程序的编译，可以把各个机型的API
做为LIB加入eclipse工程。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