

JavaME应用设计指南之多网络连接 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022_JavaME_E5_BA_94_E7_c104_144637.htm 有些设备并不允许同时打开多个网络连接，但是有的时候我们的应用程序可能需要同时打开多个连接。这样可以提高效率或者用户体验，但是应用程序应该尽量避免依赖多连接这一特性，因为对于不支持多个连接的设备你的程序可能会崩溃。解决这一问题的办法是在JAD文件中标记出设备能够同时打开的连接数量。那么如何能够得到设备支持的最大的连接数的，首先你可以去参考设备的规范。比如Motorola的设备规范中一般都列出了能够支持的连接类型，比如HTTP Socket等。以及设备能够同时打开的最大连接数。对于在规范中没有说明的那么我们只能编写一个测试用的MIDlet，先测试出这个特性。然后把他写到JAD文件中供以后使用。例如下面的JAD文件中就标记了最大连接的数量。MIDlet-1: Foo, , FooMIDlet-Jar-Size: 7063MIDlet-Jar-URL: foo.jarMIDlet-Name: FooMIDlet-Vendor: Yoyodyne Inc.MIDlet-Version: 1.0MicroEdition-Configuration: CLDC-1.1MicroEdition-Profile: MIDP-2.0Device-Max-Connection: 3 在应用程序初始化的时候可以读取这个数量，注意这个数量应该存储为全局变量，这样方便在人的地方访问。初始化的工作如下，
/* Initialization code */ String deviceMaxConnectionString = getAppProperty("Device-Max-Connection"). int deviceMaxConnection = Integer.parseInt(deviceMaxConnectionString). int openConnection

= 0.同时创建多个网路连接，如下：
if (openConnection
HttpConnection connection = (HttpConnection)
Connector.open(uri). openConnection . } else { // Fail to create
connection. } 关闭网络连接 connection.close().
openConnection--.计算连接的数量， int connection = 0. try {
while (true) { Connector.open("http://tools.ietf.org/html/rfc262"
connection). } } catch (Exception e) { // Max number reached here }
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