

# Archived:Using InputStream.skip() slows down MIDlet execution in S60 2nd and 3rd Edition (Known Issue)



Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template `{{ReviewForRemovalFromArchive|user=~~~~|write your reason here}}`.

## Overview

Using the `InputStream.skip()` method in S60 devices causes significantly slow MIDlet execution.

## Description

The `InputStream.skip(long n)` method skips and discards `n` bytes of data from the input stream.

When skipping a large amount of bytes (for example, 500000 bytes) several times with S60 devices, the method causes significantly slow MIDlet execution: with S60 3rd Edition and S60 3rd Edition, Feature Pack 1 phones this means between 5 to 10 seconds, whereas with S60 2nd Edition phones this can be even over 20 seconds. Comparison of S60 and Series 40 shows that using the `skip()` method with S60 devices is approximately 5 to 9 times (or even more) slower when skipping a large amount of bytes.

## How to reproduce

S60 and Series 40 devices are required for comparison.

The following piece of code reads the contents of a text file into the `InputStream` object. All the read content will be sent to `skip()` by using the `available()` method, and the time which the operation of the `skip()` method takes is measured. With this example, the previous procedures will be repeated 20 times by using a for-loop to visualize the slow execution.

```
InputStream instr;  
long startTime = System.currentTimeMillis();  
Form f;  
...  
for( num=0; num<20; num++){  
  
    instr=getClass().getResourceAsStream("textfile.txt");  
    instr.skip(instr.available());  
    long timePassed = System.currentTimeMillis() - startTime;  
    f.append("Round: "+(num+1)+"", Time(ms): "+timePassed+"\n");  
  
}
```

## Solution

The performance of `skip()` is improved in S60 3rd Edition, Feature Pack 2 resulting from the improved buffering when reading data from `InputStreams`.

