

new Date(), System.currentTimeMillis() and system clock accuracy

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.help/2008-07/msg00057.html>

- *From:* Numeron <irunsofastineedafinonmyhead@xxxxxxxxxxx>
 - *Date:* Thu, 3 Jul 2008 18:05:05 -0700 (PDT)
-

Hi,

Im having a problem getting a higher resolution than 10ms on my system clock (on XP) using both the getTime() method in Date and the currentTimeMillis() method in System to measure the time elapsed between two points in my code. I wrote this small peice to test this resolution:

```
long milliseconds;
for(int i = 0; i != 50; i++){
long time1 = System.currentTimeMillis();
long time2;
milliseconds = 0;
while(milliseconds < 1){
time2 = System.currentTimeMillis();
milliseconds = time2 - time1;
}
System.out.println("currentTimeMillis(): " + milliseconds + "ms");
}
for(int i = 0; i != 50; i++){
Date date1 = new Date();
Date date2;
milliseconds = 0;
while(milliseconds < 1){
date2 = new Date();
milliseconds = date2.getTime() - date1.getTime();
}
System.out.println("new Date(): " + milliseconds + "ms");
}
```

Each method assumedly gets its time from the same place since they return very much the same values (lots of 10ms and the occasional 20ms). Is there some way to update the system clock before checking it? I really need a granularity of 1ms for my program to work.

-Numeron

.