

# 1 Dalvik Debug Monitor Server: (DDMS) - Logcat

## 1.1 Sumario

- 1 Introducción
- 2 Xestión de ficheiros
- 3 Parámetros dun proceso
- 4 Parámetros do dispositivo
- 5 Enviar datos ao dispositivo
- 6 DDMS sen Eclipse
- 7 Logcat

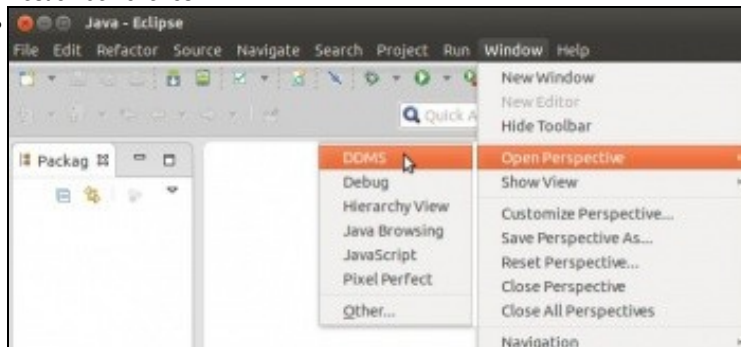
## 1.2 Introducción

- **Dalvik Debug Monitor Server (DDMS)** é unha utilidade de debug que permite visualizar o consumo de CPU, de Memoria, de rede. Tamén permite, por exemplo, enviar unha chamada ou SMS a un AVD ou coordenadas GPS para que sexan recollidas nunha aplicación.
- Úase tamén para facer debug das aplicacións como se verá na Unidade 2.
- Para o seu funcionamento precisa da utilidade ADB, e moitas das cousas realizadas con esta utilidade pódense realizar dende o DDMS.
- Co cal cando se inicia Eclipse, o DDMS inicia o servidor adb.
- No seguinte enlace pódese obter máis información: <http://developer.android.com/tools/debugging/ddms.html>
- **IMPORTANTE:** Se se está a usar a versión de Android Studio a utilidade chámase "Android Device Monitor".
  - ♦ É preciso ter instalado Java
    - ◊ No caso de Linux Java 8 ou superior.

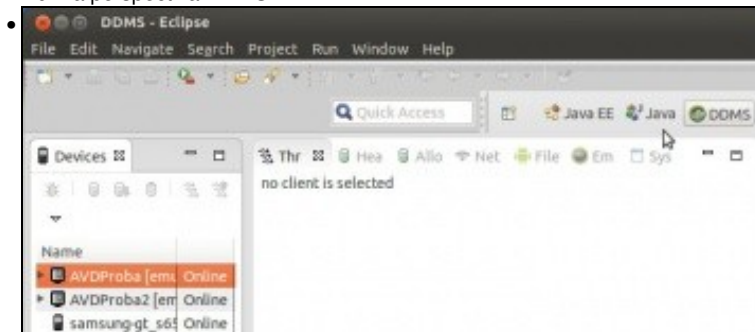
## 1.3 Xestión de ficheiros

- Ao igual que co ADB vanse poder manipular ficheiros do dispositivo pero dun modo gráfico.
- **Importante:** Se se está a usar Android Studio:
  - ♦ Abrir un proxecto.
  - ♦ Ir ao menú de: Tools-->Android-->Android Device Monitor

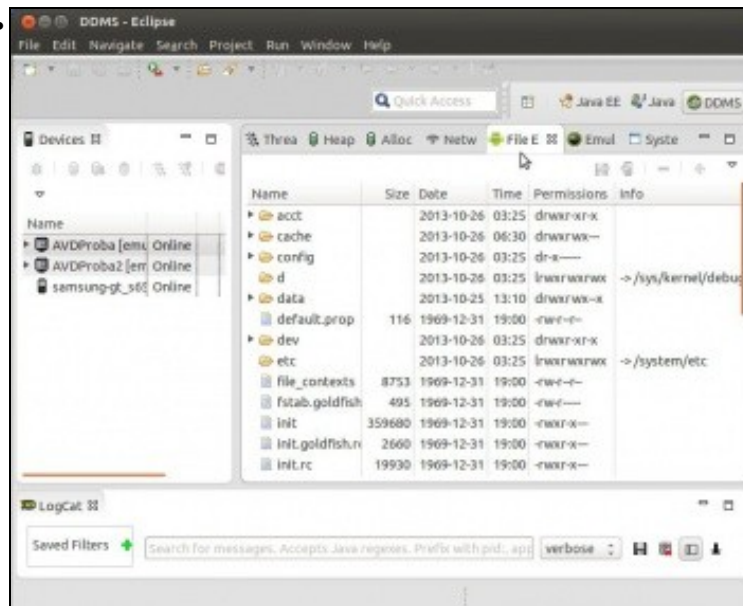
- Xestión de ficheiros



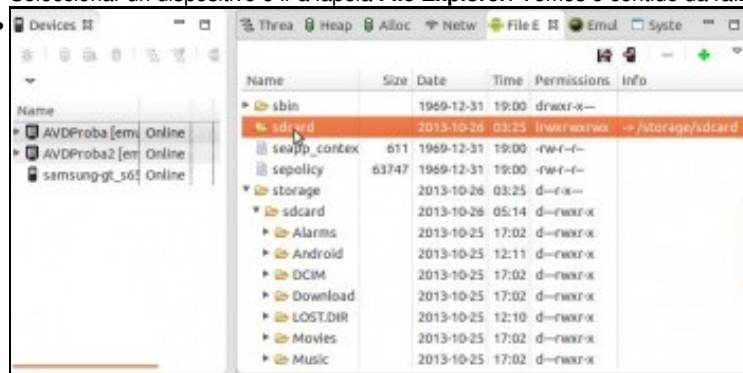
Abrir a perspectiva DDMS.



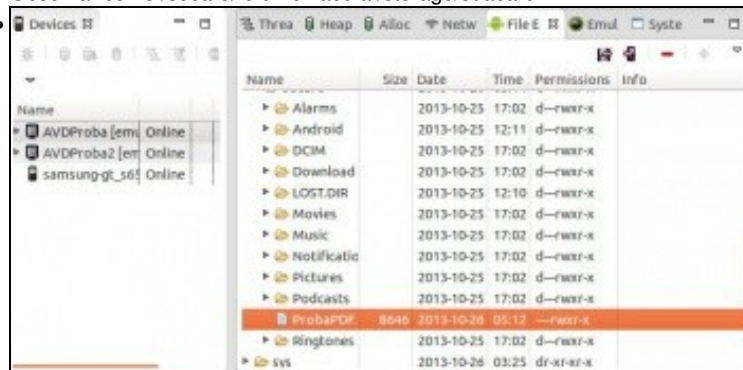
Observar os 3 dispositivos conectados: 2 AVDs e 1 físico.



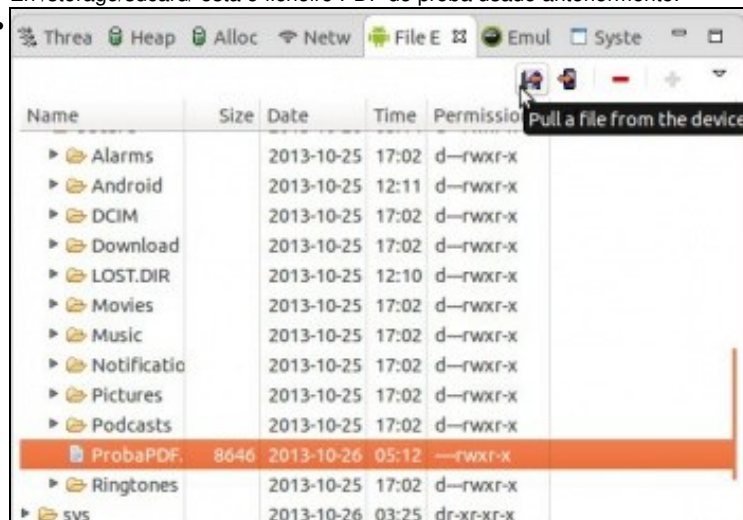
Seleccionar un dispositivo e ir á lapela **File Explorer**. Vemos o contido da raíz.



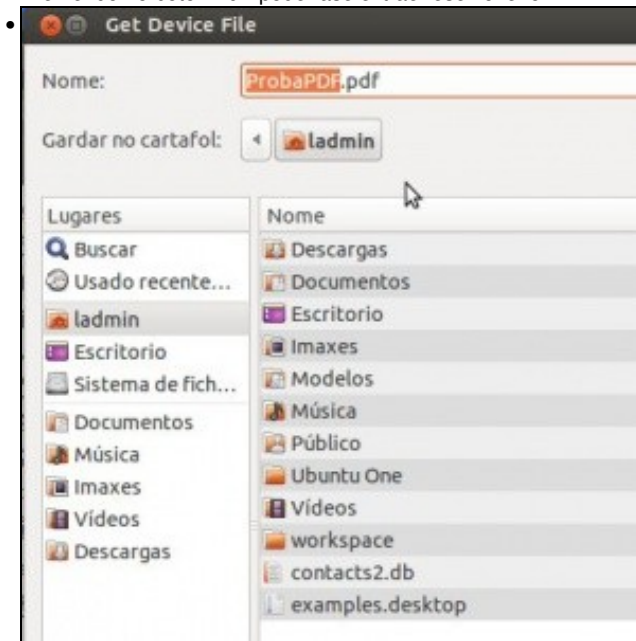
Observar como /sdcard/ é un enlace a /storage/sdcard.



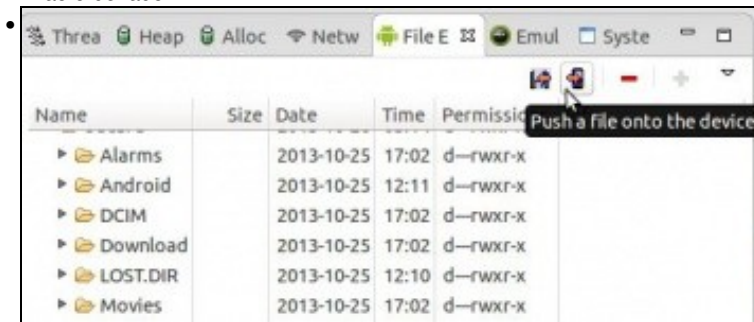
En /storage/sdcard/ está o ficheiro PDF de proba usado anteriormente.



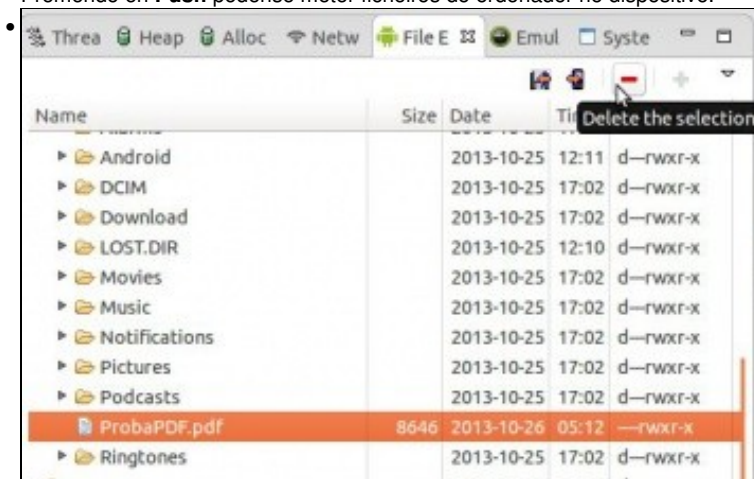
Premendo no botón **Pull** poderíase extraer ese ficheiro ...



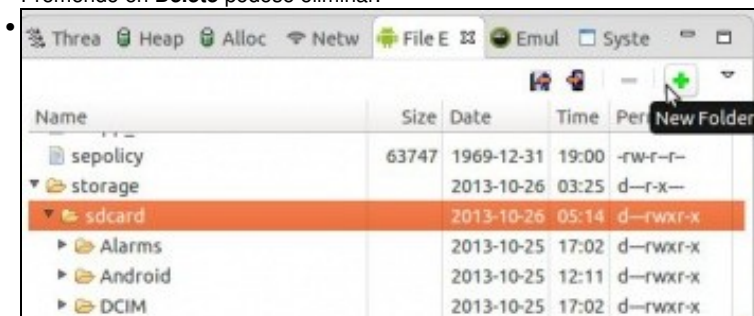
... ao ordenador.



Premendo en **Push** pódense meter ficheiros do ordenador no dispositivo.



Premendo en **Delete** pódese eliminar.



Premendo en **New folder** pódese crear un novo cartafol.

## 1.4 Parámetros dun proceso

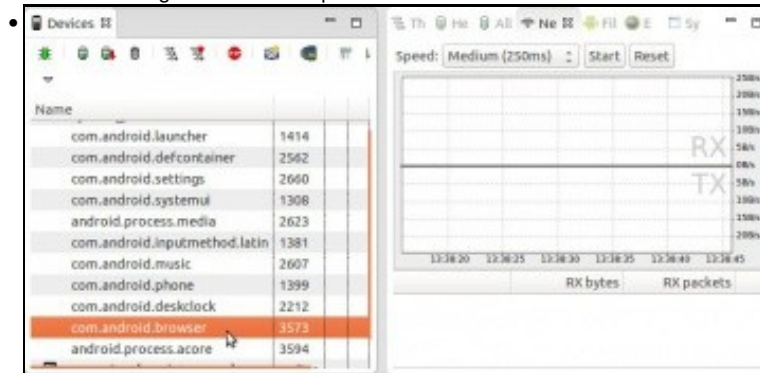
- Isto serve para comprobar os recursos que consume unha aplicación ou proceso.
- A continuación vaise lanzar o navegador web no dispositivo e vaise monitorizar.

**NOTA:** Nas imaxes INTEL vai dai un erro á hora de capturar o tráfico. En caso de querer probar esta funcionalidade baixar unha imaxe ARM dende o Android SDK Manager e crear un novo dispositivo virtual.

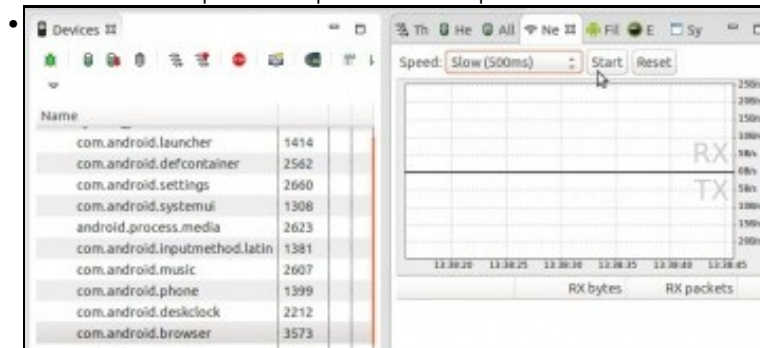
- Parámetros dun proceso



Lanzar o navegador web no dispositivo.

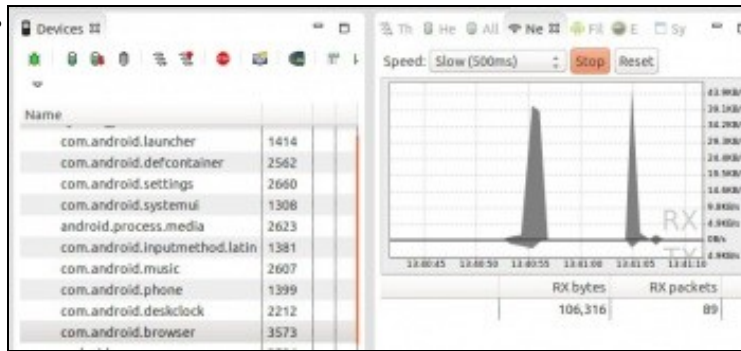


Seleccionalo no explorador de procesos e ir á lapela **Network**.

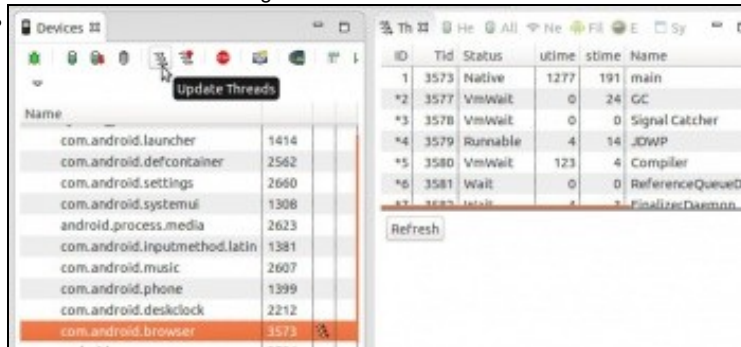


Comezar a captura de tráfico premendo en **Start**.

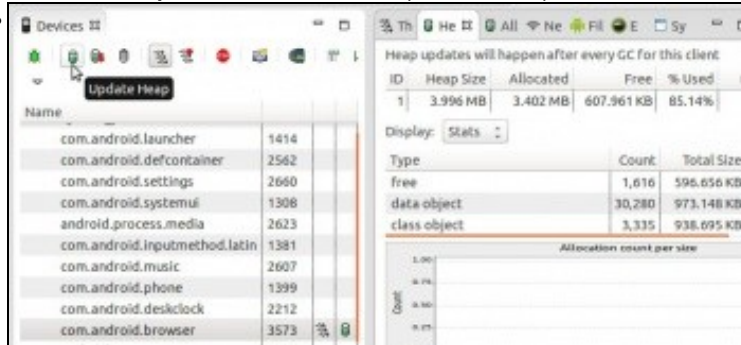




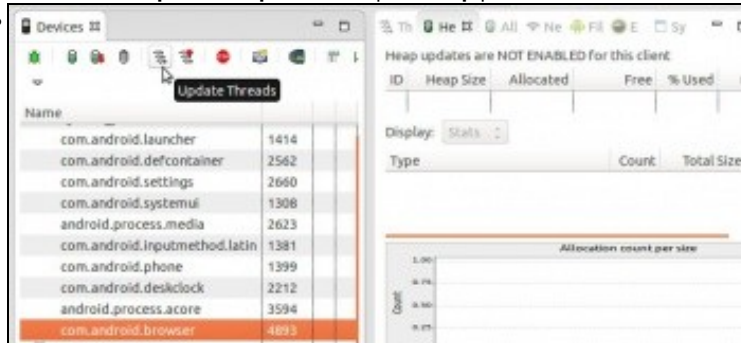
Fazer unha busca no navegador e observar o tráfico enviado e recibido.



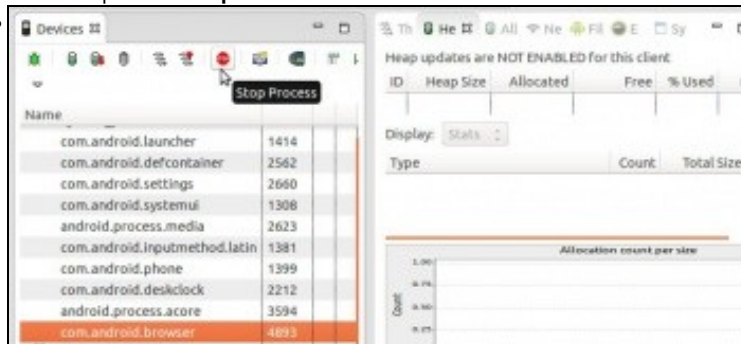
Premendo en **Update Threads** e indo á lapela **Threads** podemos ver os fíos que ten abertos ese proceso.



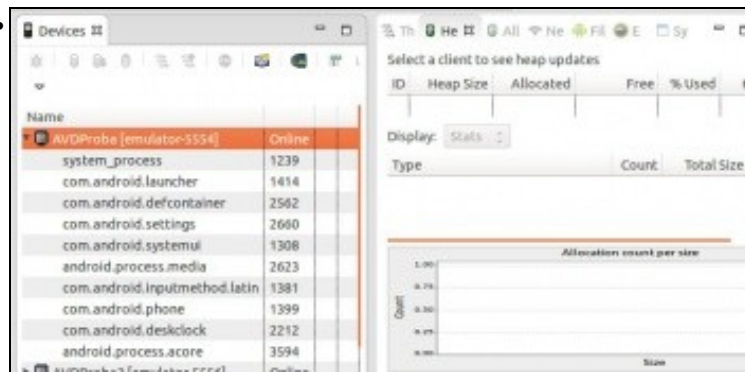
Premendo en **Update Heap** e indo á lapela **Heap** podemos ver os consumos de memoria.



Volvendo premer en **Update Threads** deshabilitamos a monitorización de threads para ese proceso. O mesmo con **Update Heap**



Podemos parar o proceso, neste caso o navegador.



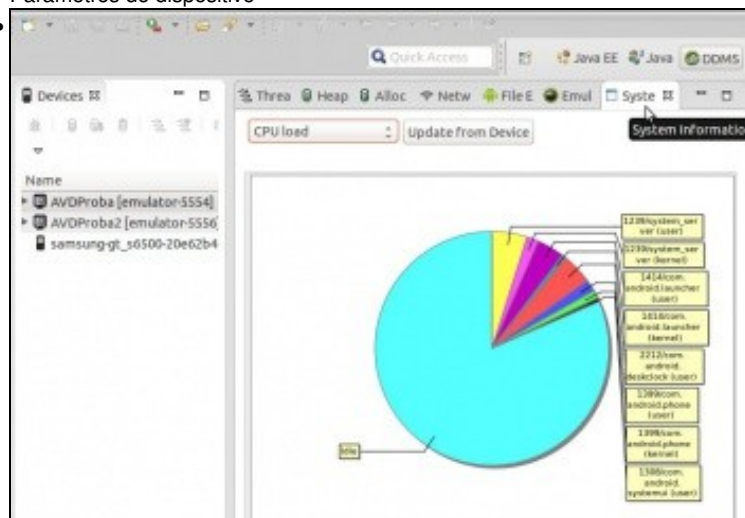
### Proceso parado ...



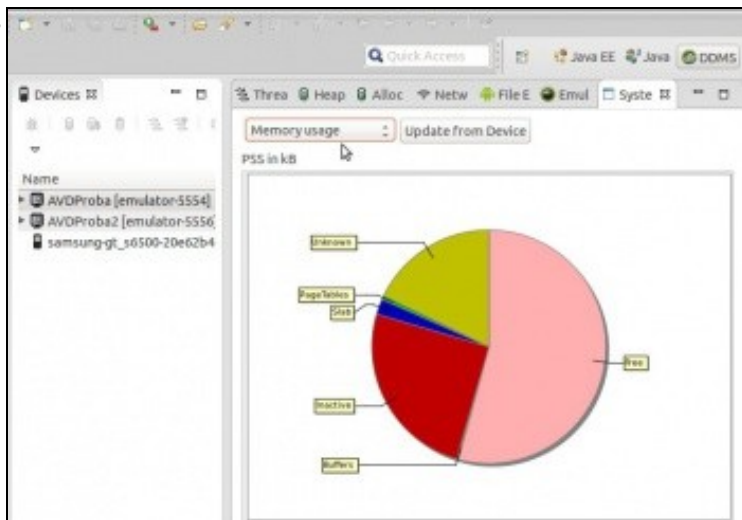
... e tamén no dispositivo.

## 1.5 Parámetros do dispositivo

- Podemos comprobar o uso de recursos que fan os procesos no dispositivo.
- Parámetros do dispositivo



Na lapela **System** podemos ver a carga de CPU do dispositivo ...

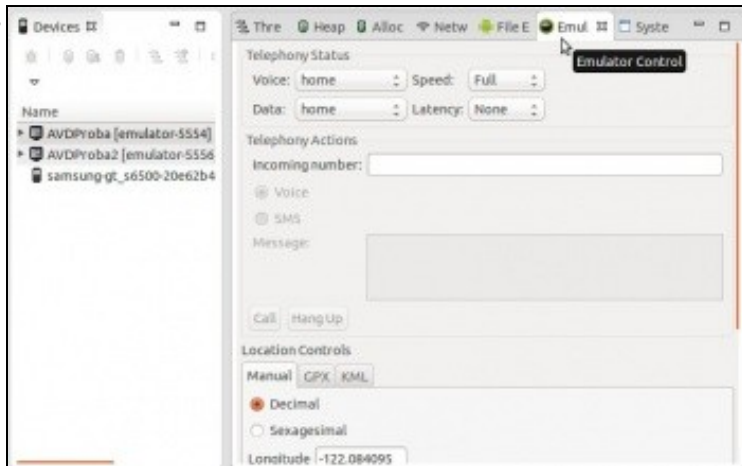


O consumo de memoria, etc.

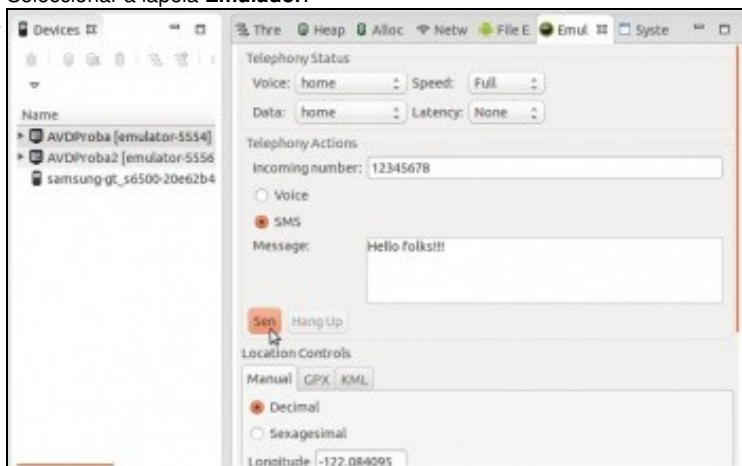
## 1.6 Enviar datos ao dispositivo

- Para poder probar as aplicacións nos AVDs ás veces é preciso que estas reciban datos do exterior: unha chamada, un sms, coordenadas gps, etc.

### • Enviar datos



Seleccionar a lapela **Emulador**.



Imos enviar un sms ao AVD. Poñer un número de teléfono, marcar **SMS**, escribir a mensaxe e premer en **Sen(d)**.

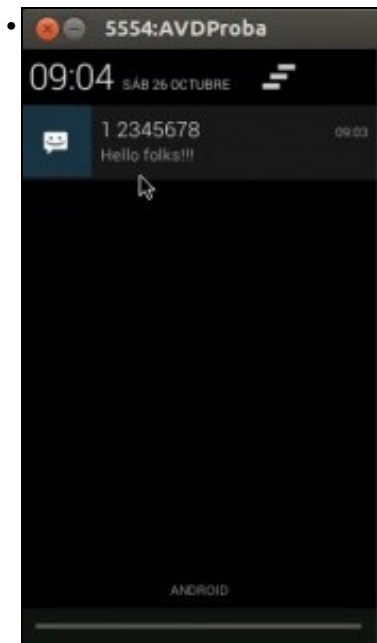


No dispositivo recíbese a notificación de SMS entrante.

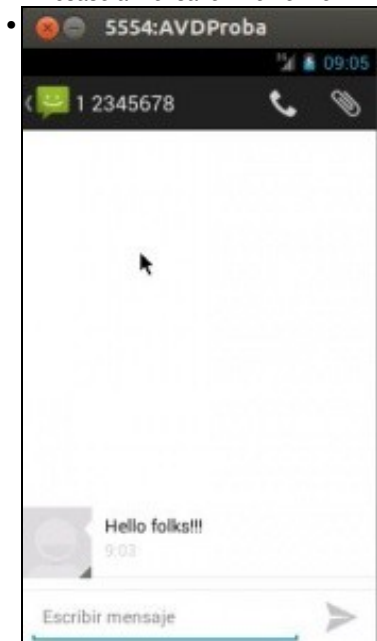


Premer na icona de aviso superior e arrastrar a barra para abaixo.

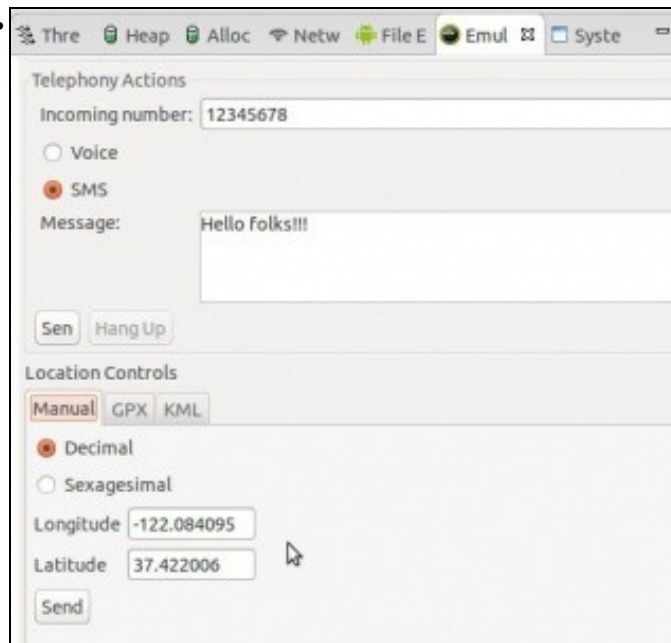




Amósase a mensaxe. Premer nel.



Seguimento da conversación.



Tamén se poderían enviar coordenadas GPS, ben de xeito manual ou ben cargadas dun ficheiro.

## 1.7 DDMS sen Eclipse

- Tamén se pode iniciar o utilidade DDMS sen facer uso de Eclipse.
- A utilidade DDMS está no cartafol **tools**.
- **Importante:** Nas novas versións do SDK para lanzar o monitor este chámase: **monitor**

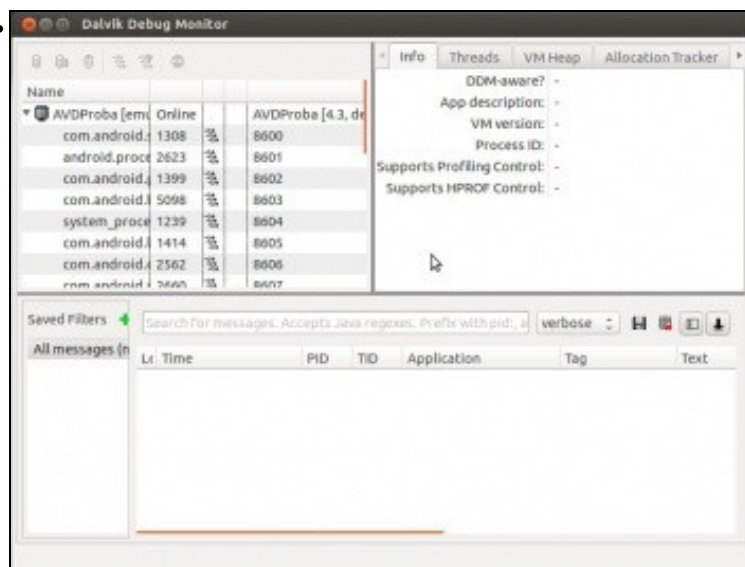
- Utilidade DDMS

```
ladmin@ubase: ~
ladmin@ubase:~$ ddms
The standalone version of DDMS is deprecated.
Please use Android Device Monitor (tools/monitor) instead.
```

Executamos **ddms** e indicanos que está obsoleta, que usemos a utilidade **monitor**.



Lanzamos **monitor**

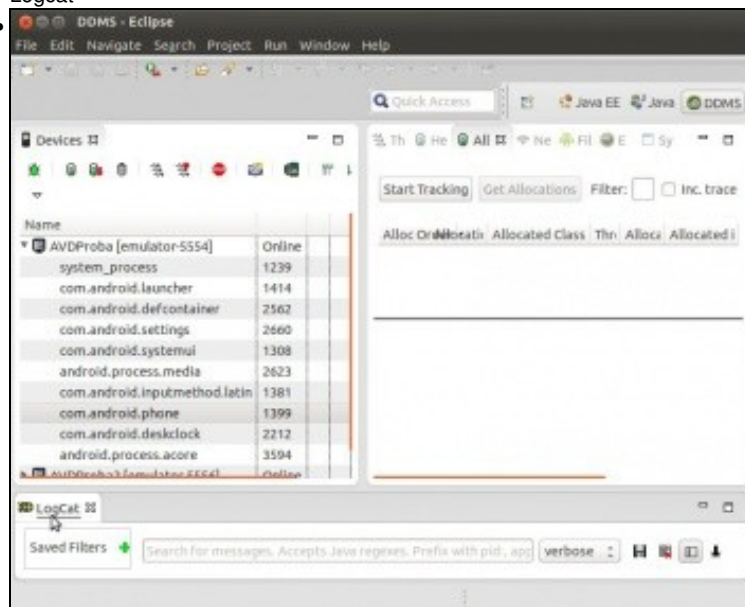


E vemos que temos a mesma ventá que cando usábamos DDMS dende Eclipse.

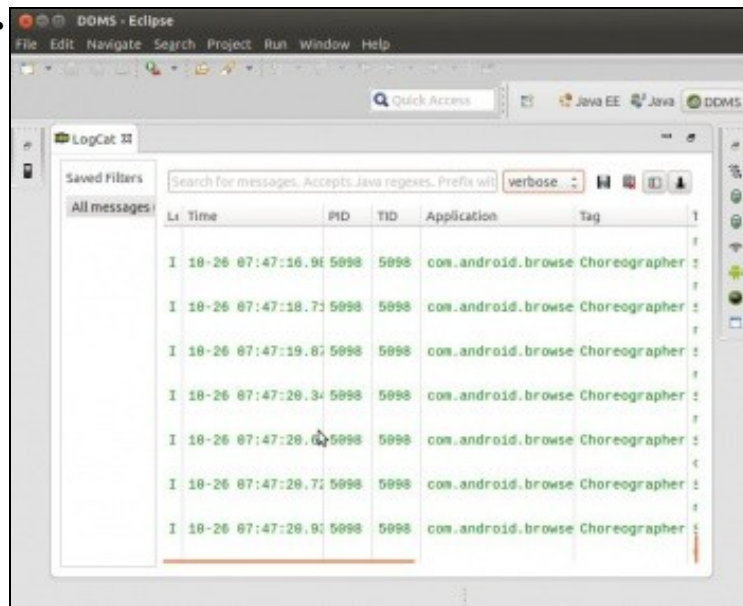
## 1.8 Logcat

- O sistema de log de Android proporciona un mecanismo polo cal vai recollendo toda a información de saída do dispositivo.
- Esa información pódese capturar e ver con **Logcat** e tamén se pode filtrar polo tipo de mensaxe.
- Na UNIDADE 2 do curso usaremos esta utilidade para controlar unha aplicación.
- Para máis información: <http://developer.android.com/tools/help/logcat.html>

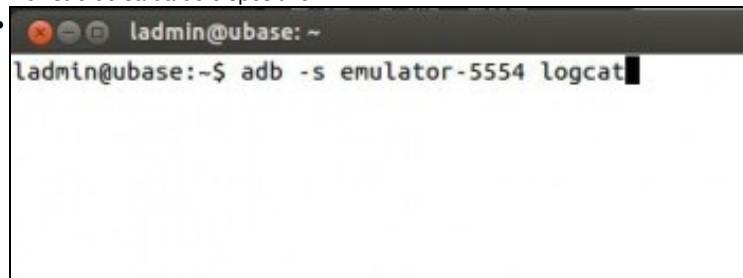
### • Logcat



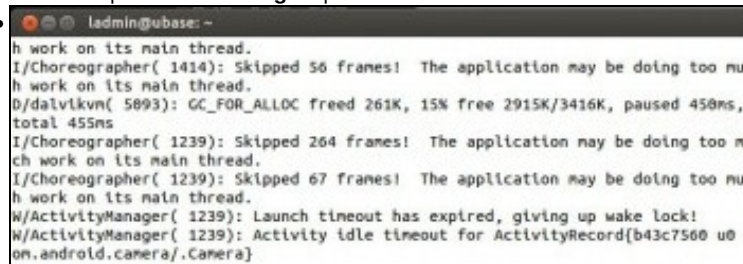
Podemos abrir a o Logcat dende unha ventá de DDMS ou Java.



Rexistro de saída do dispositivo.



Tamén se pode usar **adb logcat** para ver ...



... por consola as mensaxes que está enviando a o dispositivo.