Disassembling Android applications

Workshop

Passage en Seine III
June 17th 2011

Pierre Pronchery <khorben@defora.org>

Android: about

- Seen on http://www.dalvikvm.com/
  « Android programs are compiled into .dex (Dalvik Executable) files, which are in turn zipped into a single .apk (Android Package) file on the device. .dex files can be created by automatically translating compiled applications written in the Java programming language »
- Let's see for ourselves...

Android: just a ZIP archive #1

```
$ file app.apk
app.apk: Zip archive data, at least v2.0 to
extract
$ unzip app.apk
[...]
$ ls
AndroidManifest.xml
META-INF/
assets/
classes.dex
res/
resources.arsc
```

Android: just a ZIP archive #2

```
$ file *
AndroidManifest.xml: DBase 3 data file
META-INF:            directory
assets:              directory
classes.dex:         Dalvik dex file version 035
res:                 directory
resources.arsc:      data
$ wtf dex
wtf: I don't know what dex means!
```

Android: DEX executables

http://www.netmite.com/android/mydroid/dalvik/docs/

- DEX file layout:
  - Inspired from the DWARF file format
  - Header, identifiers, types, encodings, maps, items...
- Dalvik bytecode instruction list
- Java bytecode instruction list (for comparison)

Android: to be or not to be Java #1

- On http://en.wikipedia.org/wiki/Dalvik_virtual_machine :
« Dalvik [is] a clean-room implementation rather than a development on top of a standard Java runtime, [and] does not inherit copyright-based license restrictions from either the standard-edition or open-source-edition Java runtimes. »
« Dalvik does not align to Java SE nor Java ME class library profiles (e.g., Java ME classes, AWT or Swing are not supported). Instead it uses its own library built on a subset of the Apache Harmony Java implementation. »

Android: to be or not to be Java #2

- On http://en.wikipedia.org/wiki/Dalvik_virtual_machine : *« Unlike Java VMs, which are stack machines, the Dalvik VM is a register-based architecture:*
  - *The VM was slimmed down to use less space*
  - *The constant pool has been modified to use only 32-bit indexes to simplify the interpreter*
  - *Standard Java bytecode executes 8-bit stack instructions. Local variables must be copied [...] by separate instructions. Dalvik [uses a] 16-bit instruction set that works directly on local variables. The local variable is commonly picked by a 4-bit 'virtual register' field. »*

Android: Dalvik bytecode

```
$ cat src/arch/dalvik.ins
[...]
{ "add-double",          0xab,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "add-double/2addr",    0xcb,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
{ "add-float",           0xa6,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "add-float/2addr",     0xc6,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
{ "add-int",             0x90,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "add-int/2addr",       0xb0,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
{ "add-int/lit8",        0xd8,   OP1F, OP_REG8,      OP_REG8,      OP_U8        },
{ "add-int/lit16",       0xd0,   OP1F, OP_REG4,      OP_REG4,      OP_U16       },
{ "add-long",            0x9b,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "add-long/2addr",      0xbb,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
{ "aget",                0x44,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-boolean",        0x47,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-byte",           0x48,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-char",           0x49,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-object",         0x46,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-short",          0x4a,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "aget-wide",           0x45,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "and-int",             0x95,   OP1F, OP_REG8,      OP_REG8,      OP_REG8      },
{ "and-int/2addr",       0xb5,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
{ "and-int/lit8",        0xdd,   OP1F, OP_REG8,      OP_REG8,      OP_U8        },
{ "and-int/lit16",       0xd5,   OP1F, OP_REG4,      OP_REG4,      OP_U16       },
{ "and-so/muchmore",     0xc0,   OP1F, OP_REG4,      OP_REG4,      AOT_NONE     },
```

Get the code: what and where

- Disassembler found in the DeforaOS asm project
- Hosted by and part of DeforaOS
- Development happens in a CVS tree:

```
$ cvs
-d:pserver:anonymous@anoncvs.defora.org:/Data/C
VS co DeforaOS
```

- Formal releases available
- Web interface and daily archives available
- asm depends on libSystem and libcpp

Get the code: formal releases

Fresh from yesterday:

- http://www.defora.org/os/download/download/3527/libSy
- http://www.defora.org/os/download/download/3542/cpp-
- http://www.defora.org/os/download/download/3545/asm-

Simple as pie hopefully:

```
$ make PREFIX=”/usr/local” install
[...]
```

(may install a conflicting cpp binary; just remove it)

Get the code: compilation

```
$ make bootstrap
[...]
The source tree is now configured for your
environment. Essential libraries and tools will
now be installed in this folder: "/usr/local"
You can still exit this script with the CTRL+C
key combination.
Otherwise, press ENTER to proceed.
  CTRL+C
$ (cd System/src/libSystem && make install)
$ (cd Apps/Devel/src/cpp && make install && rm -f
/usr/local/cpp)
$ (cd Apps/Devel/src/asm && make install)
```

Use the code: disassemble

```
$ deasm
Usage: deasm [-a arch][-f format] filename
       deasm [-a arch][-f format] -s string
       deasm -l

$ deasm classes.dex
classes.dex: dex-dalvik

Disassembly of section .text:

[...]
```

Modify the code: contributing

Through DeforaOS:

- Introduce yourself on devel@lists.defora.org
- Report bugs or wishes
- Send patches
- Gain my trust (and an account)
- Commit directly :)

Android: Google did it

Some links to share:

- http://mylifewithandroid.blogspot.com/
- http://pallergabor.uw.hu/androidblog/
- http://developer.android.com/reference/dalvik/bytecode/C
- http://zeaster.blogspot.com/2007/11/how-to-decompile-d
- http://www.dalvikvm.com/

Suggestions

- Anything I may have missed?
- Feedback, <3 and !<3 mail at khorben@defora.org