

Atom Badge
photo
All weekly snippets from Andy McFadden (fadden)

PDB page - WWW page - MOMA - OKRs - Google Resume - Email - This page with tabs
Wed, Jan 2, 2008
[delete]

Last week:

- (Holiday vacation)

This week:

- Work on mterp.

Thu, Dec 27, 2007
[delete]

Last week:

- Worked on Dalvik VM:
- Worked on "mterp".
- Corrected reflection access check on protected fields.
- Did some more directory renaming.
- One interview candidate lunch.

This week:

- (Christmas holiday + two vacation days)

Mon, Dec 17, 2007
[delete]

Last week:

- Worked on Dalvik VM:
- Split DEX file access code out into a static library. Shifted lots of code around.
- Added ability to run "dexdump" on a jar/apk/zip with "classes.dex" inside.
- Added "dexlist" utility to list methods found in a DEX archive.
- Added API for Dalvik-level instruction counting.
- Changed internal exception throws to use "descriptor" form of class name.
- Added additional access check on protected field access through reflection.
- Stripped out some more simulator cruft.
- Started learning Python.

This week:

- Work on asm version of interpreter.

Mon, Dec 10, 2007
[delete]

Last week:

- Finished up filename remapping in simulator.
- Added "fake" device entry for /sys/android_power entries.
- Removed last bit of sim-specific stuff in libhardware.
- Added touch-screen support to simulator.
- Added system property to enable Java language assertions in Dalvik.
Added support for class/package name matching in -ea/-da.
- Fixed a bug in Dalvik that could try to execute code in an uninitialized class when throwing an exception from JNI.
- Started on Dalvik instruction count reporting.

This week:

- Finish up instruction count mechanism.
- Start on asm interpreter for Dalvik?

Mon, Dec 3, 2007
[delete]

Last week:

- Worked on Dalvik VM:
- Fixed a bug with multiple re-initializations of a class that fails to load.
- Fixed a bug with null references in reflection code.
- Corrected float-to-int conversions.
- Investigated "can't create 764 threads" issue.
- Created an alternative implementation of system properties for simulator.
- Worked on transparent filename remapping in simulator.
- Renamed a couple of major directories in the source tree.
- (one interview)

This week:

- Finish sim filename remapping.

UNITED STATES DISTRICT COURT NORTHERN DISTRICT OF CALIFORNIA
TRIAL EXHIBIT 27
CASE NO. 10-03561 WHA
DATE ENTERED _____
BY _____
DEPUTY CLERK

- Work on Dalvik stuff.

Mon, Nov 26, 2007
[delete]

Last week:

- Got basic set of fake drivers (display, input, logging) working with simulator. Also implemented "vibrating notification mechanism".
- Removed a bunch of sim-specific code.
- Fixed a bug in JNI local references.
- Chased down an incorrect exception thrown from native code when "inflate" sees bad data.
- (Out Thursday/Friday for Thanksgiving)

This week:

- Continue work on sim, notably system properties and filename remapping.

Mon, Nov 19, 2007
[delete]

Last week:

- Celebrated the public launch of the Android SDK.
- Read an unhealthy number of blogs, news articles, and press releases.
- Worked on simulator evolution:
 - Moved simulator communication channel out of runtime and into preload lib.
 - Got "fake device" infrastructure working.
 - Got most of /dev/graphics/fb0, /dev/tty0, and /dev/input/events0 working.
- Chased after the "Binder thread explosion" bug.

This week:

- Finish up /dev/input/events0.
- Write /dev/log/*.
- Start on filename substitution.

Mon, Nov 12, 2007
[delete]

Last week:

- Reviewed patent application.
- Removed various incarnations of the "J word" from the SDK.
- Began simulator overhaul: launch runtime in separate xterm, with LD_PRELOAD wrapper. Removed "ui-hardware" classes and interfaces.
- Fixed "ls -R" behavior.
- Fixed bug in dmtrace (nee jtrace) profiling.
- Various minor changes to prep for SDK launch.

This week:

- Continue simulator work.

Mon, Nov 5, 2007
[delete]

Last week:

- Spent some time configuring a MacPro as a Linux build machine.
- Worked with jparks and mathias to get the Linux simulator working again.
- Stripped out some simulator cruft.
- Started fiddling around with planned simulator updates.
- Last-minute stuff for M3.

This week:

- Continue work on simulator evolution.
- Read lots of news articles.

Mon, Oct 29, 2007
[delete]

Last week:

- Added binary event log output to "logcat". Fixed the binary logging implementation. Added a log tag map.
- Worked on Dalvik VM:
 - Reduced overhead when calling into the interpreter.
 - Rearranged EXPORT_PC handling so we do less work.
 - Made the main VM thread more generic, so that it's possible for other threads to join() on the main thread.
 - Fixed bytecode verification when finding the best superclass of two array classes.
 - Show more detail when a finalizer throws an uncaught exception.
 - Ensure existing processes blow up if you replace one of the core Java classes.
- Added bytecode disassembly to "dexdump" tool.

This week:

- Continue with Dalvik stuff.
- Look into making the simulator build more device-like.

Mon, Oct 22, 2007

[delete]

Last week:

- Resurrected stand-alone benchmark test app.
- Worked on Dalvik VM:
 - Improved verification of invoke-super. Removes need for a runtime test.
 - Changed verification to switch off of a property, rather than an #ifdef.
 - Reduced thread-local storage lookups by passing a pointer around in performance-critical areas. Sped up JNI and synchronized blocks.
- Pushed for (and got) a decision on how we're fixing our IPC mechanism.
- Participated in a promotion & slotting committee (29 candidates).

This week:

- Continue Dalvik performance & correctness changes.

Mon, Oct 15, 2007

[delete]

Last week:

- Wrote up proposal for Binder driver replacement. Discussed with a few people.
- Helped track down some JNI global reference leaks.
- Pursued the "slow classloader" problem (identified, partially addressed with a ZipEntry change, redirected to core java lib).
- Pursued the "uncaught exceptions don't work" problem (identified, redirected to framework).
- Worked on Dalvik VM:
 - Refactored the "instanceof" cache into a more general mechanism.
 - Used the cache to speed up interface calls.
 - Correctly handle virtual calls to abstract methods (for out-of-sync Jar files).
- Updated some developer docs.

This week:

- Promotion/slotting committee.
- Continue with Dalvik stuff.
- Get IPC replacement figured out.

Mon, Oct 8, 2007

[delete]

Last week:

- Added feature in VM that copies stdout/stderr to log file.
- Made a couple minor changes to help others debug problems.
- Switched to IPC rewrite full time.
 - Wrote some code for the previous design (not checked in).
 - Became concerned about scheduling and started thinking about alternate plans.

This week:

- Finalize what we're doing with IPC.

Mon, Oct 1, 2007

[delete]

Last week:

- Worked on Dalvik VM:
 - Added some static tests to Java verifier. Rearranged failure logging to reduce output size.
 - Enabled the Java bytecode verifier.
 - Generalized the reference table mechanism. Use the "expandable" mode for JNI global refs and some GC heap stuff.
 - On thread detach, release all monitors entered via the JNI MonitorEnter function.
 - For JNI compliance, fail if somebody tries to detach a thread with Java frames on the stack.
 - Added JNI GetObjectRefType method. We should be JNI 1.6 compliant now.
- Followed up on digit's TCP_NODELAY discovery by running some experiments and then disabling Nagle in DDMS and VM JDWP.
- Spent some time with crazybob talking about the IPC implementation.
- Helped some people figure out issues uncovered with -Xcheck:jni.
- Followed up on a couple of pioneer issues ("bad" APK hosing boot process, ServerSocket not closing properly).
- Finished peer reviews.

This week:

- Investigate interface call speedup.
- Clean up some stuff in the VM.
- Help out with memory leak issues.

Mon, Sep 24, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Implemented some slower but EABI-correct register-access primitives in interpreter.
 - Implemented a "fast path" in the ARM native JNI call bridge to make things faster under EABI.
 - Fixed initialization of java.lang.Class.
 - Tweaked thread dumps to help with system debugging.
- Fixed some minor stuff in Java libs.
- Forced "check JNI" to be enabled. Dealt with the ensuing hilarity.
- Chased after a couple of memory leaks.
- Read through some "new IPC" stuff from crazybob.
- Worked on performance peer reviews (six).
- Mentor for Andy Stadler.
- (Got caught up after vacation)

This week:

- Finish peer reviews.
- Work with crazybob on "new IPC".
- Deal with whatever comes up.

Mon, Sep 17, 2007
[delete]

Last week:

- (out on vacation)

This week:

- Get caught up on current events.
- Enable verification.
- Write static verification tests.
- Work on perf review stuff.

Tue, Sep 4, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Made "-Xverify:{none,remote,all}" do what they should. Got verification working end-to-end.
 - Rearranged optimization to respect "-Xdexopt:{all,verified}".
- Added some code to force optimization+verification of class files when the system first boots. This reduces the delay seen on first app launch.
- Worked on performance review.
- (Out Friday)

This week:

- Monday was holiday, in Tuesday, away Wednesday-Friday. Mostly trying not to break things.
- Out all next week too.

Mon, Aug 27, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Rearranged class loading to perform verification on non-pre-verified classes.
 - Enabled "lite" verification; code flow analysis still too slow.
 - Corrected handling of package-scope access checks during opt/verify.
 - Added page access guards on LinearAlloc regions to help track down an EABI memory corruption bug.
 - Expanded set of JNI calls you're allowed to make with an exception pending (1.6 spec).
 - Updated jni.h to remove some fancy new warnings.
 - Added flags to disable signal catching.
- Partially restored the stand-alone performance tests.
- Took some measurements with and without mkf's SSA optimizations.
- Spent some time examining the performance of JDWP to the device and emulator.
- Attended the Android coming-out tech talk.

This week:

- Work on performance review.

- Improve performance of verified code flow analysis.
- Add some static verification tests.

Mon, Aug 20, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Finished development on bytecode structural verifier. Added more detailed checks on short integers. Corrected common superclass determination. Reduced memory usage and improved speed.
 - Blocked optimization and pre-verification of classes defined in multiple DEX files.
 - Eliminated VMThrowable class, folding it into Throwable.
 - Added logging of exceptions thrown from finalizers.
 - Added printLoadedClasses to help with Zygote performance examination.
 - Fixed a bug that would cause Eclipse to break on caught exceptions that straddled a native call.

This week:

- Continue static verification work. Enable (warn-only mode) in live system.

Mon, Aug 13, 2007
[delete]

Last week:

- (away on vacation)

This week:

- Get caught up.
- Finish bytecode verification.
- Address some memory profiling analysis needs.

Mon, Aug 6, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Continued work on verifier. Code-flow analysis is complete, but we've decided to extend it slightly beyond what the spec requires.
 - Fiddled with exception logging to help track down excessive exception activity.
 - Aligned array objects for EABI.

This week:

- Out on vacation all week.

Mon, Jul 30, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Made good progress on bytecode verifier.
 - Moved uncaught exception handling for the main thread into the thread-detach code.
 - Fixed Class.newInstance to do access checks properly (Noser).
 - Fixed some reflection exception handling problems exposed by jython.
- One phone screen.

This week:

- Continuing on with the verifier.

Mon, Jul 23, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Worked on bytecode verifier.
 - Changed the way internal exceptions are initialized to match expected behavior.
 - Added JNI vfprintf() and abort() hooks.
 - Fixed up some stuff from the Noser integration.
- Filed several bugs on improper use of JNI NewGlobalRef in our sources.
- One interview.

This week:

- Continue work on verifier.

Mon, Jul 16, 2007

[delete]

Last week:

- Worked on Dalvik VM:
 - Continued work on Java bytecode verifier
 - Found last week's heisenbug (gcc-4.2.0 bug)
- Worked with others to chase misc bugs (cheese-scrolling throw-fest, spinning Spinner, Bitmap heap trasher, sluggish Mac sim, nuance DetachCurrentThread)
- (away on vacation Friday)

This week:

- Continue Java bytecode verifier

Mon, Jul 9, 2007

[delete]

Last week:

- Worked on Dalvik VM:
 - Worked on some classfile verification stuff.
 - Added -Xverify option.
 - Changed thread startup from spin-wait to condition wait. Took some thread startup timings.
 - Enabled `__memcmp16()` and fixed some bugs in `String.compareTo()`.
- Removed JamVM from the build.
- Spent a bunch of time chasing a Heisenbug (771329).
- (Wednesday was a holiday)

This week:

- Continue chasing the Heisenbug.
- Work on Java class verification.

Mon, Jul 2, 2007

[delete]

Last week:

- Dalvik VM:
 - Various improvements to JDWP debug support (simplified startup, "server=n").
 - Fixed a bug in stack overflow exception handling.
 - Changed frame depth calculation to improve single-step execution performance.
 - Added some additional tests to `-Xcheck:jni`.
- Fixed a stall in DDM initialization.
- Fiddled with `LD_ASSUME_KERNEL` for our goobuntu systems.
- Worked on my OKRs.

This week:

- Start work on Java classfile verifier.

Mon, Jun 25, 2007

[delete]

Last week:

- Worked on Dalvik VM:
 - Finished EABI-compliant JNI call bridge (in ARM assembly).
 - Aligned long/double fields on 64-bit boundaries by reshuffling field order.
 - Wrote replacement 64-bit arg decoder for internal natives.
 - Fixed a rare bug that hit when a debugger attached at the wrong moment.
 - Fixed a bug that could lock up a VM if DDMS was left connected for too long.
 - Reduced class lookup-by-name calls by caching a reference to `[Foo in Foo`.
 - Experimented with a JDWP speed improvement (which didn't work).
- Fixed an fd leak in DDMS.
- One interview.

This week:

- Work on Dalvik.
- Figure out some OKRs.

Mon, Jun 18, 2007

[delete]

Last week:

- Worked on Dalvik VM:
 - Reduce native heap usage in Zip archive support.
 - Fixed some VM startup ordering issues that were blocking Noser.
 - Wrote a simple linear allocator to hold bits and pieces of class files.
 - Made the VM-internal hash table load factor adjustable, and tweaked it for better performance.
 - Allocate Java thread stacks with `mmap()` instead of `malloc()`.
 - Started work on ARM EABI support.

- Chased down some JNI problems for Nuance that turned out to be a bug in our dynamic linker.
- One interview.

This week:

- Finish ARM EABI support in Dalvik.

Mon, Jun 11, 2007
[delete]

Last week:

- Added ZipFileRO, an extremely memory-conscious Zip archive access class.
- Replaced all device-side usage of ZipFile/ZipEntry classes with ZipFileRO.
- Evaluated correctness & performance of mathias' __memcpy16.
- Remove system class loader from app loader call sequence.
- Worked on Dalvik VM:
 - Implemented java.lang.reflect.Method.getExceptionTypes(), now that we have exception lists in DEX.
- One phone screen.
- (Out Thursday/Friday on vacation)

This week:

- Continue with some Dalvik memory footprint reduction and performance stuff.

Mon, Jun 4, 2007
[delete]

Last week:

- Moved to a new office.
- Worked on Dalvik VM:
 - Improved spin-on-suspend handling.
 - Added support for Java thread priorities. Added missing pieces of Thread/ThreadGroup.
 - Added AccessController and ProtectionDomain features, notably the getStackDomains() method, for Noser security work.
 - Explored alternative stack allocation methods.
 - Fixed a JDWP bug.
- Two interviews (one cancelled shortly before it started).
- (Monday was holiday)

This week:

- Work on reduced-heap Zip support.
- (On vacation Thursday/Friday)

Tue, May 29, 2007
[delete]

Last week:

- Worked on DalvikVM:
 - Sped up native calls.
 - Wrote inline versions of String.equals() and String.compareTo() (6-40x faster than Java).
 - Changed DEX file section alignment to 64-bit for pending EABI changes.
 - Added "-Xss" flag, use it to set default stack sizes.
 - Rearranged the wait-for-suspend code.
- Added TRACE_ALLOC_COUNTS flag to jtrace. If set, the allocation counters are written into the key file.
- Added some misc fields (total calls, syscall benchmark) to jtrace key header.
- Met with patent lawyers.
- One interview.

This week:

- Dalvik VM performance & correctness.

Mon, May 21, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Experimented with fixed register assignments in interpreter core.
 - Tracked down and fixed a bug in debugger method invocation.
 - Added java.lang.Thread constructor that takes a stack size.
- Corrected size bounds in VM.
 - In DEX optimizer, replace calls to Object.<init> with instructions that are treated as no-ops in standard interpreter.
 - Added an "inline implementation" instruction. Wrote inline versions of String.length() and String.charAt().
- Substantially rewrote "efficient Java" document. Reformatted entire document, edited existing sections, added new material.

This week:

- Continue performance work.

Mon, May 14, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Enabled instruction rewriting in the DEX optimizer. Improves the speed of virtual method calls and accesses to instance fields.
 - Wrote a cache for the results of "instanceof" and "checkcast" operations. Iterated until useful.
 - Moved DEX class name lookup from a heap-allocated hash table to shared-memory storage.
 - Changed a large number of functions from "JValue blah(...)" to "void blah(..., JValue* pResult)". This works out better on ARM, and improves the performance of native methods returning void (of which we have many).
 - Added some JNI performance and correctness tests.
 - Replaced the JNI native call bridge with a hand-coded ARM assembly version.
- Continued the quest for Java readability.
- One phone screen.

This week:

- Jury duty. Assuming I'm not called up:
 - Continue Dalvik performance work.
 - Add some instrumentation features requested by jparks/veenstra/etc.

Mon, May 7, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Completed work on optimization of virtual fields and instance methods.
 - Reworked optimization mechanism to use fork+exec instead of just fork.
 - Added exception-wrapping to java.lang.reflect.Method.invoke implementation.
 - Added an instruction counter to get a sense for which instructions are popular and which are going unused.
 - Made some minor performance tweaks.

This week:

- Continue Dalvik performance work.
- Finish Java readability review work.

Mon, Apr 30, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Reorganized DEX file loading so that generation and caching of the byte-swapped version happens generally (i.e. not merely as a side-effect of Jar extraction).
 - Compute and store inter-DEX dependency information. Re-optimize files when a dependency changes.
 - Examined VM loaded-classes hash table performance.
 - Fixed a problem with debugger-initiated method invocations that throw exceptions.
- Worked on DDMS:
 - Warp client selection when a VM is waiting for a debugger.

This week:

- Finish up Dalvik field/method access optimizations.

Mon, Apr 23, 2007
[delete]

- Worked on Dalvik VM:
 - Removed most of the overhead associated with debugger support from the interpreter core, resulting in a ~30% speedup.
 - Made miscellaneous minor speedups (strict-aliasing, interpreter size reduction).
 - Tried some performance enhancements that didn't work.
 - Made some improvements to the performance tests.

Mon, Apr 16, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Spent a bunch of time fiddling with the performance tests to try to get a consistent set of meaningful numbers.
 - Improved performance by inlining part of class/field/method/string

resolution.

- Moved some low-use values out of the Method struct. Reduced heap usage by 12 bytes per method.
- Reduced the situations in which we initialize a class to those specified in the VM spec.
- Reshuffled VM source tree, moving headers next to sources.
- Show debug info breakdown in "dexdump -S".
- Fixed a bug that prevented strings in Object arrays from displaying nicely in the debugger.
- Worked on DDMS:
 - Changed "Halt" button to call Runtime.halt() instead of Runtime.exit(). Bypasses some of the exit processing.
- Cleaned up some code for a readability review.
- Wrote up OKRs.

This week:

- Continue Dalvik VM performance work.

Mon, Apr 9, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Rewrote DEX file loading to maximize use of shared memory, reducing native heap requirements.
 - Fix a spin-on-suspend when a GC happens while a profiled thread is waiting to exit.
- Worked on DDMS:
 - Cosmetic touch-ups on heap stats.
 - Added a "cause GC" button.
 - Added space for native heap info.
- Provided a work-around for slow log window performance under Mac OS.
- Wrote a command-line Java performance measurement tool.
- Attended daily "aardvark" release meetings.

This week;

- Continue performance improvements, focusing on DEX file optimization work.

Mon, Apr 2, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Added "inflate" support to classes.dex loading.
 - Investigated debugger performance issues.
 - Corrected VM internal hash table implementation.
 - Quick-reject debugger connection attempts when one is already connected.
 - When waiting for a debugger, return after it has connected and been idle for a set period (to allow it a chance to set breakpoints).
- Worked on DDMS:
 - Show tid, status, utime, and stime in the "threads" panel. Update periodically.
 - Chased after some adb-related problems.
 - Reviewed and updated developer documentation.
- Participated in "aardvark" status meetings.
- One phone screen.

This week:

- Work on Dalvik VM

Mon, Mar 26, 2007
[delete]

Last week:

- Worked on "ddms"
 - Finished hooking up "preferences" items.
 - Added a color key to the heap map.
 - Don't try to scan the device if it's in the boot loader.
 - Started work on retrieving per-thread CPU usage stats.
- Worked on Dalvik VM
 - Prefer new pthread_cond_timedwait_monotonic() for Object.wait().
 - Allow global references table to expand.
 - Implemented JNI direct byte buffer calls.
 - Reduced the VM's debug-level log output.
 - Implemented java.lang.reflect.Field get/set calls.
 - Fiddled with some VM options after the first ever Dalvik performance review. Toned down paranoia on "release" builds to improve device performance.

Dalvik is now the default on devices.

This week:

- Finish per-thread CPU stats in ddms.
- Dalvik VM performance work.

Mon, Mar 19, 2007

[delete]

Last week:

- Work on DDMS
 - Added preferences screens with JFace. Implemented most of them.
 - Improved the network I/O.
 - Refactored client list. Added "waiting for debugger" icon.
 - Send the app name up when it changes.
 - Fixed some ordering issues when running Dalvik on the device.
- Spent a day hunting down edge cases exposed by attaching "jdb" and letting the system run.

This week:

- Finish wiring up the DDMS preferences.
- Work on Dalvik VM (monotonic clock waits, global ref table).

Mon, Mar 12, 2007

[delete]

Last week:

- Worked on DDMS:
 - Added it to the build. Got it working on Macs.
 - Added "debug selected VM" feature.
 - Added skeletal heap panel UI for dbort.
 - Changed reply routing to allow for generic "fail" packet.
 - Added device screen shot feature.
 - Added device command output capture; used with "dumpstate", "dumpsys", and "logcat".
 - Fixed lots of little things in the UI.
- Added Java "assert" statement support to both VMs.
- In Dalvik VM: ignore breakpoints and other events while executing code on behalf of the debugger.
- In Dalvik VM: treat JNI warnings as fatal errors (with a to-be-added command-line option to mellow them out). Made "Check JNI" the simulator default.
- Spent some time at the "Android boot camp" dry run.
- One interview.

This week:

- Continue work on DDMS.

Mon, Mar 5, 2007

[delete]

Last week:

- Added thread creation/death notifications to DDM.
- Started work on DDM server user interface. Spent most of the week on this.
- Added an "is debugger connected?" feature and a "change JDWP listen port" feature to both VMs.
- Participated in "client team sourcing event".
- One interview.

This week:

- Continue work on DDM.

Mon, Feb 26, 2007

[delete]

Last week:

- Worked on DDM:
 - Got enough working to register Java "hello" message handlers on both sides and exchange data.
 - Updated debugger pass-through to recycle client connection when debugger drops.
 - Added DDMS support to JDWP code.
 - Added a couple of calls to our NIO implementation.
- Helped reed/abliss chase after some resource leaks that presented as global reference table overflows.
- One phone screen
- (Monday was holiday)

This week:

- Continue with DDM server and client.

Tue, Feb 20, 2007

[delete]

Last week:

- Fixed a number of bugs in Dalvik and JamVM.
- Added file info to "jtrace" output.
- Worked on DDMS. Currently shuffling packets around with NIO.
- One phone screen.

This week:

- Continue work on DDMS.

Mon, Feb 12, 2007
[delete]

Last week:

- Implemented per-thread and default uncaught exception handlers (in both JamVM and Dalvik VM).
- Worked on Dalvik VM:
 - Added support for -Xrunjdpw/-agentlib:jdpw= options.
 - Rearranged JDWP startup to allow "suspend=y".
- Worked on Dalvik Debug Monitor design doc.
- Started on Dalvik Debug Monitor server (ddms). Added SWT to main build.

This week:

- Continue work on ddms.

Mon, Feb 5, 2007
[delete]

Last week:

- Worked on Dalvik VM:
 - Fixed bug in libffi that only manifested on Linux/x86 with optimizations enabled.
 - Implemented parts of ClassLoader.getResource().
 - Added DEX-in-Jar support to user-defined class loaders.
 - Corrected JNI AttachCurrentThread behavior.
 - Corrected JNI FindClass behavior to use system class loader rather than bootstrap. This allows CLASSPATH to be used for the initial class.
 - Check method return types in -Xcheck:jni.
 - Chased after misc bugs people found now that Dalvik is more widely used.
 - Added a few things that the "mauve" test suite wanted.

This week:

- Implement "-Xrunjdpw" args.
- Get ClassLoader.getResource() working.
- Continue to address Dalvik issues as they arise.

Mon, Jan 29, 2007
[delete]

Last week:

- Work on Dalvik VM:
 - Added support for loading DEX files from Zip archives to bootstrap class loader.
 - Added local-reference abuse checking to -Xcheck:jni. Fixed a couple of problems it found, reported others.
 - Added libffi, so we can use PowerPC macs.
- Announced general availability of Dalvik at group meeting.
- (Out sick most of Friday)

This week:

- Work on Dalvik VM.
- Work on resource loading.

Mon, Jan 22, 2007
[delete]

Last week:

- Dalvik VM:
 - Finished work on initating class loaders in Dalvik.
 - Replaced the hold-all "u8" type with a union throughout the code.
- Corrected some bugs in the JDWP implementation in both VMs. Added some missing features that IntelliJ required.
- Spent most of Tuesday with visitors from Korea.
- (Monday was holiday)

This week:

- Roll Dalvik VM out to the soon-to-be adoring masses.

Tue, Jan 16, 2007

[delete]

Last week:

- Wrote a JNI calling convention converter. Ran it against our sources, which used C-style JNI calls in C++ code.
- Dalvik VM:
 - Reworked some VM class loader innards.
 - Did most of the work required to support initiating class loaders.
- Perf: write peer reviews.
- One interview.

This week:

- Finish initiating class loaders.
- Meet with some Java people.
- Work on Dalvik.

Mon, Jan 8, 2007

[delete]

Last week:

- More jni.h tweaks, mostly to make JNIHelp.h work with either C or C++.
- Changed JamVM's RegisterNatives method to link methods immediately, rather than on first use.
- Override the class loader in JNI FindClass while handling JNI_OnLoad.
- Fixed a couple of bugs in the JDWP support.
- Work on Dalvik VM:
 - Added basic class loader support to VM. Added DEX file loading to Java ClassLoader.
 - Added "lazy" JNI native method binding by signature.
 - Hooked up local variable table support for the debugger.
 - Improved debugger object registry (fewer registrations, more appropriate data structure).
- One interview.
- (four day week)

This week:

- Performance reviews.
- Work on Dalvik VM.