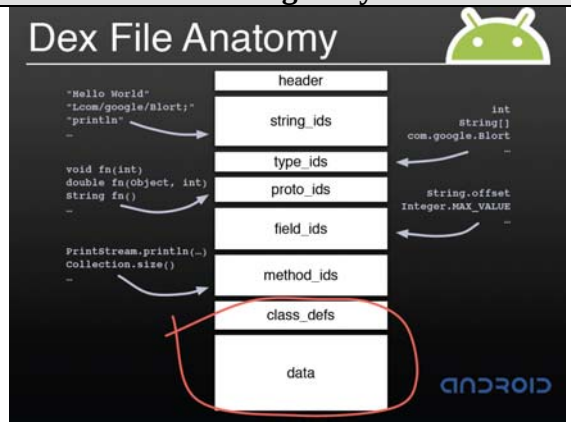
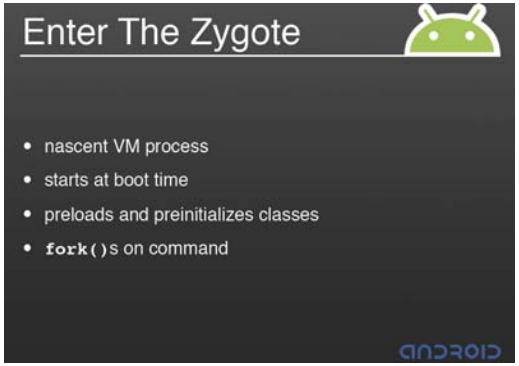
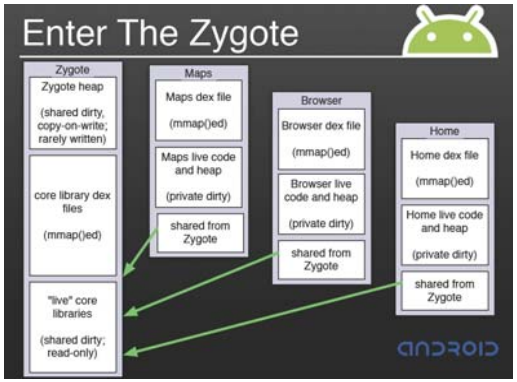
The ’720 Patent	Infringed By
	 (Dalvik Presentation, Slide 13) Corresponding Dalvik Video at 6:39: “And then towards the bottom there are a series of class definitions. So a dex file contains multiple classes….” *See, e.g.*, dalvik\vm\analysis\DexPrepare.c (in Gingerbread) *See, e.g.*, dalvik\vm\analysis\DexOptimize.c. (in earlier versions) */* ** Return the fd of an open file in the DEX file cache area. If the cache* ** file doesn’t exist or is out of date, this will remove the old entry,* ** create a new one (writing only the file header), and return with the* ** ”new file” flag set.* *** **…** ** On success, the file descriptor will be positioned just past the ”opt”* ** file header, and will be locked with flock. ”*pCachedName” will point* ** to newly-allocated storage.* **/* *int dvmOpenCachedDexFile(const char* fileName, const char* cacheFileName,u4 modWhen, u4 crc, bool isBo otstrap, bool* pNewFile, bool createIfMissing)* *{* *int fd, cc;*

The ’720 Patent	Infringed By
	struct stat fdStat, fileStat; *bool readOnly = false;* **pNewFile = false;* *retry:* */** ** Try to open the cache file. If we’ve been asked to,* ** create it if it doesn’t exist.* **/* *fd = createIfMissing ? open(cacheFileName, O_CREAT|O_RDWR, 0644) : -1;* *if (fd < 0) {* *fd = open(cacheFileName, O_RDONLY, 0);* *if (fd < 0) {* *if (createIfMissing) {* *LOGE(“Can’t open dex cache ’%s’: %s\n”,* *cacheFileName, strerror(errno));* *}* *return fd;* *}* *readOnly = true;* *}* … *}*
6. A system according to claim 1, further comprising: a process cloning mechanism to instantiate the child runtime system process by copying the memory space of the master runtime system process into a separate memory space for the child runtime system process.	Android includes a process cloning mechanism to instantiate a child runtime system process by copying the memory space of a master runtime system process into a separate memory space for the child runtime system process. *See*

The ’720 Patent	Infringed By
	 (Android Presentation, Slide 82) Corresponding Android Video at 44:30: “The init process starts up a really neat process called zygote….It uses copy-on-write to maximize re-use and minimize footprint so that data structures are shared and it won’t do a full copy unless some of those data structures are to be modified.” *See also*  (Dalvik Presentation, Slide 25) Corresponding Dalvik Video at 13:48:

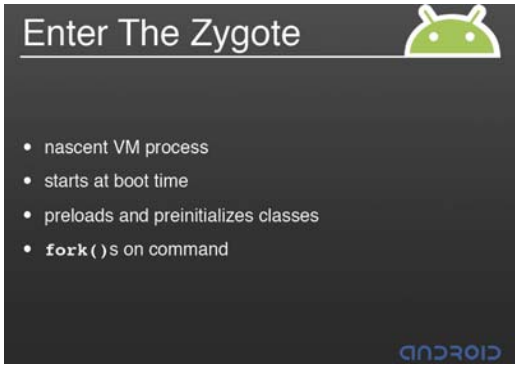
The ’720 Patent	Infringed By
	“What we do with the zygote, as its name implies,…when it gets a command to start up a new application, it does a normal Unix fork and then that child process becomes that target application. And the result of that is this.”  (Dalvik Presentation, Slide 26) Corresponding Dalvik Video at 14:40: “So the zygote, again, has made, has made this heap of objects, it’s made this live dex structure and then each application that then starts up, instead of having its own memory for those things, it just shares it with the zygote and also with any other app that’s also on the system.” *See also* http://developer.android.com/guide/basics/what-is-android.html. “Android Runtime …The Dalvik VM relies on the Linux kernel for underlying functionality such as threading and low-level memory management. Linux Kernel Android relies on Linux version 2.6 for core system services such as security, memory management, process management, network stack, and driver model. The kernel also acts as

The ’720 Patent	Infringed By
	an abstraction layer between the hardware and the rest of the software stack.” *See also*, Lowe, Robert, Linux Kernel Process Management, April 15, 2005. Sample Chapter is provided courtesy of Sams, http://www.informit.com/articles/article.aspx?p=370047&seqNum=2&rll=1. “Copy-on-Write …In Linux, fork() is implemented through the use of copy-on-write pages. Copy-on-write (or COW) is a technique to delay or altogether prevent copying of the data. Rather than duplicate the process address space, the parent and the child can share a single copy. The data, however, is marked in such a way that if it is written to, a duplicate is made and each process receives a unique copy.” Example source code files in libcore\dalvik\src\main\java\dalvik\system\Zygote.java, dalvik\vm\native\dalvik_system_Zygote.c, linux-2.6\kernel\fork.c, external\kernel-headers\original\linux\sched.h. *See, e.g.*, libcore\dalvik\src\main\java\dalvik\system\Zygote.java. */*** ** Forks a new Zygote instance, but does not leave the zygote mode.* ** The current VM must have been started with the -Xzygote flag. The* ** new child is expected to eventually call forkAndSpecialize()* *** ** @return 0 if this is the child, pid of the child* ** if this is the parent, or -1 on error* **/* *native public static int fork();* */***

The ’720 Patent	Infringed By
	** Forks a new VM instance. The current VM must have been started* ** with the -Xzygote flag. NOTE: new instance keeps all* ** root capabilities. The new process is expected to call capset().* *** ** @param uid the UNIX uid that the new process should setuid() to after* ** fork()ing and and before spawning any threads.* ** @param gid the UNIX gid that the new process should setgid() to after* ** fork()ing and and before spawning any threads.* ** @param gids null-ok; a list of UNIX gids that the new process should* ** setgroups() to after fork and before spawning any threads.* ** @param debugFlags bit flags that enable debugging features.* ** @param rlimits null-ok an array of rlimit tuples, with the second* ** dimension having a length of 3 and representing* ** (resource, rlim_cur, rlim_max). These are set via the posix* ** setrlimit(2) call.* *** ** @return 0 if this is the child, pid of the child* ** if this is the parent, or -1 on error.* **/* *native public static int forkAndSpecialize(int uid, int gid, int[] gids,* *int debugFlags, int[][] rlimits);* *See, e.g.*, dalvik\vm\native\dalvik_system_Zygote.c. */* native public static int forkAndSpecialize(int uid, int gid,* ** int[] gids, int debugFlags);* **/* *static void Dalvik_dalvik_system_Zygote_forkAndSpecialize(const u4* args, JValue* pResult)* *{* *pid_t pid;* *pid = forkAndSpecializeCommon(args);* *RETURN_INT(pid);* *}* … */** ** Utility routine to fork zygote and specialize the child process.* **/*

The ’720 Patent	Infringed By
	static pid_t forkAndSpecializeCommon(const u4 args, bool isSystemServer)* *{* *pid_t pid;* *uid_t uid = (uid_t) args[0];* *gid_t gid = (gid_t) args[1];* *ArrayObject* gids = (ArrayObject *)args[2];* *u4 debugFlags = args[3];* *ArrayObject *rlimits = (ArrayObject *)args[4];* *int64_t permittedCapabilities, effectiveCapabilities;* *if (isSystemServer) {* */* ** Don't use GET_ARG_LONG here for now. gcc is generating code* ** that uses register d8 as a temporary, and that's coming out* ** scrambled in the child process. b/3138621* **/* *//permittedCapabilities = GET_ARG_LONG(args, 5);* *//effectiveCapabilities = GET_ARG_LONG(args, 7);* *permittedCapabilities = args[5] | (int64_t) args[6] << 32;* *effectiveCapabilities = args[7] | (int64_t) args[8] << 32;* *} else {* *permittedCapabilities = effectiveCapabilities = 0;* *}* *if (!gDvm.zygote) {* *dvmThrowException("Ljava/lang/IllegalStateException;",* *"VM instance not started with -Xzygote");* *return -1;* *}* *if (!dvmGcPreZygoteFork()) {* *LOGE("pre-fork heap failed\n");* *dvmAbort();* *}* *setSignalHandler();*

The ’720 Patent	Infringed By
	dvmDumpLoaderStats("zygote"); *pid = fork();* *if (pid == 0) {* *int err;* */* The child process */* **….** *} else if (pid > 0) {* */* the parent process */* *}* *return pid;* *}* *See, e.g.*, linux-2.6\kernel\fork.c. */* ** Ok, this is the main fork-routine.* *** ** It copies the process, and if successful kick-starts* ** it and waits for it to finish using the VM if required.* **/* *long do_fork(unsigned long clone_flags,* *unsigned long stack_start,* *struct pt_regs *regs,* *unsigned long stack_size,* *int __user *parent_tidptr,* *int __user *child_tidptr)* *{* *struct task_struct *p;* *int trace = 0;* *long nr;* **…** *p = copy_process(clone_flags, stack_start, regs, stack_size,* *wake_up_new_task(p, clone_flags);* **…** *tracehook_report_clone_complete(trace, regs,* *clone_flags, nr, p);*

The ’720 Patent	Infringed By
	… *return nr;* *}*
7. A system according to claim 1, wherein the master runtime system process is caused to sleep relative to receiving the process request.	Android includes a master runtime system process that is caused to sleep relative to receiving a process request. *See*  (Dalvik Presentation, Slide 25) Corresponding Dalvik Video at 13:48: “What we do with the zygote, as its name implies, …it sort of sits on a socket and it waits for commands….” *See also*

The ’720 Patent	Infringed By

	
	(Android Presentation, Slide 82)
	
	Corresponding Android Video at 44:25:
	“The init process starts up a really neat process called zygote….And so zygote is a nascent VM process that initializes a Dalvik VM and preloads a lot of its libraries and it forks on request to create new VM instances for managed processes….”
	
	See, e.g., base\core\java\com\android\internal\os\ZygoteConnection.java.
	
	*/**
	* Constructs instance from connected socket.
	*
	* @param socket non-null; connected socket
	* @throws IOException
	*/
	ZygoteConnection(LocalSocket socket) throws IOException {
	mSocket = socket;
	mSocketOutStream
	= new DataOutputStream(socket.getOutputStream());
	mSocketReader = new BufferedReader(
	new InputStreamReader(socket.getInputStream()), 256);
	mSocket.setSoTimeout(CONNECTION_TIMEOUT_MILLIS);
	try {
	peer = mSocket.getPeerCredentials();*

The ’720 Patent	Infringed By
	} catch (IOException ex) { *Log.e(TAG, “Cannot read peer credentials”, ex);* *throw ex;* *}* *}* */*** ** Returns the file descriptor of the associated socket.* *** ** @return null-ok; file descriptor* **/* *FileDescriptor getFileDesciptor() {* *return mSocket.getFileDescriptor();* *}* */*** ** Reads start commands from an open command socket.* ** Start commands are presently a pair of newline-delimited lines* ** indicating a) class to invoke main() on b) nice name to set argv[0] to.* ** Continues to read commands and forkAndSpecialize children until* ** the socket is closed. This method is used in ZYGOTE_FORK_MODE* *** ** @throws ZygoteInit.MethodAndArgsCaller trampoline to invoke main()* ** method in child process* **/* *void run() throws ZygoteInit.MethodAndArgsCaller {* *int loopCount = ZygoteInit.GC_LOOP_COUNT;* *while (true) {* **…** *if (runOnce()) {* *break;* *}* *}* *}* */*** ** Reads one start command from the command socket. If successful,* ** a child is forked and a {@link ZygoteInit.MethodAndArgsCaller}* ** exception is thrown in that child while in the parent process,*

The ’720 Patent	Infringed By
	** the method returns normally. On failure, the child is not* ** spawned and messages are printed to the log and stderr. Returns* ** a boolean status value indicating whether an end-of-file on the command* ** socket has been encountered.* *** ** @return false if command socket should continue to be read from, or* ** true if an end-of-file has been encountered.* ** @throws ZygoteInit.MethodAndArgsCaller trampoline to invoke main()* ** method in child process* **/* *boolean runOnce() throws ZygoteInit.MethodAndArgsCaller {* *String args[];* *Arguments parsedArgs = null;* *FileDescriptor[] descriptors;* *try {* *args = readArgumentList();* *descriptors = mSocket.getAncillaryFileDescriptors();* *} catch (IOException ex) {* *Log.w(TAG, “IOException on command socket “ + ex.getMessage());* *closeSocket();* *return true;* *}* *if (args == null) {* *// EOF reached.* *closeSocket();* *return true;* *}* … *int pid;* … *pid = Zygote.forkAndSpecialize(parsedArgs.uid, parsedArgs.gid,* *parsedArgs.gids, parsedArgs.debugFlags, rlimits);* *} catch (IllegalArgumentException ex) {* *logAndPrintError (newStderr, “Invalid zygote arguments”, ex);* *pid = -1;* *} catch (ZygoteSecurityException ex) {* *logAndPrintError(newStderr,* *“Zygote security policy prevents request: “, ex);* *pid = -1;*

The ’720 Patent	Infringed By
	} *if (pid == 0) {* *// in child* *handleChildProc(parsedArgs, descriptors, newStderr);* *// should never happen* *return true;* *} else { /* pid != 0 */* *// in parent...pid of < 0 means failure* *return handleParentProc(pid, descriptors, parsedArgs);* *}* *}* … */*** ** Reads an argument list from the command socket/* ** @return Argument list or null if EOF is reached* ** @throws IOException passed straight through* **/* *private String[] readArgumentList()* *throws IOException {* */*** ** See android.os.Process.zygoteSendArgsAndGetPid()* ** Presently the wire format to the zygote process is:* ** a) a count of arguments (argc, in essence)* ** b) a number of newline-separated argument strings equal to count* *** ** After the zygote process reads these it will write the pid of* ** the child or -1 on failure.* **/* *int argc;* *try {* *String s = mSocketReader.readLine();* *if (s == null) {* *// EOF reached.* *return null;* *}* *argc = Integer.parseInt(s);* *} catch (NumberFormatException ex) {* *Log.e(TAG, “invalid Zygote wire format: non-int at argc”);* *throw new IOException(“invalid wire format”);*

The '720 Patent	Infringed By
	} **…** *String[] result = new String[argc];* *for (int i = 0; i < argc; i++) {* *result[i] = mSocketReader.readLine();* *if (result[i] == null) {* *// We got an unexpected EOF.* *throw new IOException("truncated request");* *}* *}* *return result;* *}*
8. A system according to claim 1, wherein the object-oriented program code is written in the Java programming language.	Android includes object-oriented program code that is written in the Java programming language. *See* Google I/O 2010 Video, entitled "A JIT Compiler for Android's Dalvik VM," presented by Ben Cheng and Bill Buzbee (Google's Android Team), available at http://developer.android.com/videos/index.html#v=Ls0tM-c4Vfo ("JIT Video") at time 1:58. "Now, if you are going to write a program for Android, you are most likely going to write it in the Java programming language and then push the source code through the SDK. And what pops out at the end is an executable targeted to the Dalvik virtual machine." *See also* http://developer.android.com/guide/basics/what-is-android.html. "What is Android? Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language."
10.pre. A method for dynamic preloading of classes through memory space cloning of a master runtime system process, comprising:	Android and the Android SDK include methods for performing the steps described in the claim. *See* claim 1.pre.

The ’720 Patent	Infringed By
10.a. executing a master runtime system process;	*See* claim 1.c.
10.b. obtaining a representation of at least one class from a source definition provided as object-oriented program code;	*See* claim 1.c.
10.c. interpreting and instantiating the representation as a class definition in a memory space of the master runtime system process; and	*See* claim 1.d.
10.d. cloning the memory space as a child runtime system process responsive to a process request and executing the child runtime system process;	*See* claim 1.e.
10.e. wherein cloning the memory space as a child runtime system process involves instantiating the child runtime system process by copying references to the memory space of the master runtime system process into a separate memory space for the child runtime system process; and	*See* claim 1.f.
10.f. wherein copying references to the memory space of the master runtime system process defers copying of the memory space of the master runtime system process until the child runtime system process needs to modify the referenced memory space of the master	*See* claim 1.f.

The ’720 Patent	Infringed By
runtime system process.	
11. A method according to claim 10, further comprising: determining whether the instantiated class definition is available in a local cache associated with the master runtime system process.	*See* claim 2.
12. A method according to claim 11, further comprising: locating the source definition if the instantiated class definition is unavailable in the local cache.	*See* claim 3.
13. A method according to claim 10, further comprising: resolving the class definition.	*See* claim 4.
14. A method according to claim 10, further comprising: maintaining the source definition as a class file on at least one of a local and remote file system.	*See* claim 5.
15. A method according to claim 10, further comprising: instantiating the child runtime system process by copying the memory space of the master runtime system process into a separate memory space for the child runtime system process.	*See* claim 6.
16. A method according to claim 10, further comprising: causing the master runtime system process to sleep relative to receiving the	*See* claim 7.

The ’720 Patent	Infringed By
process request.	
17. A method according to claim 10, wherein the object-oriented program code is written in the Java programming language.	*See* claim 8.
19. A computer-readable storage medium holding code for performing the method according to claim 10.	The Accused Instrumentalities include storage devices that store, distribute, or run code for Android or the Android SDK. They encompass a computer-readable storage medium holding code for performing the method according to claim 10. *See* claim 10.
20.pre. An apparatus for dynamic preloading of classes through memory space cloning of a master runtime system process, comprising:	The Accused Instrumentalities include devices that run Android or the Android SDK. An Android-based device is an apparatus. *See* claim 1.pre.
20.a. A processor;	*See* claim 1.a.
20.b. A memory	*See* claim 1.b.
20.c. means for executing a master runtime system process;	*See* claim 1.c. *See also, e.g.,* ’720 patent, 8:14-17, 38-44, FIGs. 2, 6, 7: “The method 100 [FIG. 6] is described as a sequence of process operations or steps, which can be executed, for instance, by the runtime environment 31 of FIG. 2 or other components.” “FIG. 7 is a flow diagram showing the routine 120 for loading a master JVM process 33 for use in the method 100 of FIG. 6. One purpose of the routine is to invoke the master JVM process 33 and to preload classes into the prewarmed state 41 for inheritance by cloned JVM processes 34. Initially, the master JVM process 33 begins execution at device boot time (block 121).”
20.d. means for obtaining a representation of at least one class from a source definition provided as object-oriented program code;	*See* claim 1.c. *See also, e.g.,* ’720 patent, 6:46-54, FIGs. 2, 10: “A set of core Java foundation classes is specified in a bootstrap class loader 39 and

The '720 Patent	Infringed By
	application classes in a system application class loader 40. Class loading requires identifying a binary form of a class type as identified by specific name, as further described below with reference to FIG. 10. Depending upon whether the class was previously loaded or referenced, class loading can include retrieving a binary representation from source and constructing a class object to represent the class in memory."
20.e. means for interpreting and means for instantiating the representation as a class definition in a memory space of the master runtime system process; and	*See* claim 1.d. *See also, e.g.*, '720 patent, 6:61-67, FIG. 2: "The master JVM process 33 invokes the bootstrap class loader 39 and system application class loader 40 for every class likely to be requested by the applications. Thus, the prewarmed state 41 includes the class loading for applications prior to actual execution and the initialized and loaded classes are inherited by each cloned JVM process 34 as the inherited prewarmed state 42."
20.f. means for cloning the memory space as a child runtime system process responsive to a process request and means for executing the child runtime system process;	*See* claim 1.e. *See also, e.g.*, '720 patent, 5:33-37, FIG. 2: "The runtime environment 31 executes an application framework that spawns multiple independent and isolated user application process instances by preferably cloning the memory space of a master runtime system process."
20.g. wherein the means for cloning the memory space is configured to clone the memory space of a child runtime system process using a copy-on-write process cloning mechanism that instantiates the child runtime system process by copying references to the memory space of the master runtime system process into a separate memory space for the child runtime system process and that defers copying of the	*See* claim 1.f. *See also, e.g.*, '720 patent, 6:12-19, FIGs. 2, 5A, 5B: "When implemented with copy-on-write semantics, the process cloning creates a logical copy of only the references to the master JVM process context. Segments of the referenced master JVM process context are lazily copied only upon an attempt by the cloned JVM process to modify the referenced context. Therefore as long as the cloned JVM process does not write into a memory segment, the segment remains shared between parent and child processes."

The ’720 Patent	Infringed By
memory space of the master runtime system process until the child runtime system process needs to modify the referenced memory space of the master runtime system process.	
21. A system according to claim 1, further comprising: a resource controller to set operating system level resource management parameters on the child runtime system process.	Android includes a resource controller to set operating system level resource management parameters on the child runtime system process. *See, e.g.*, libcore\dalvik\src\main\java\dalvik\system\Zygote.java. */** * Forks a new VM instance. The current VM must have been started * with the -Xzygote flag. NOTE: new instance keeps all * root capabilities. The new process is expected to call capset(). * * @param uid the UNIX uid that the new process should setuid() to after * fork()ing and and before spawning any threads. * @param gid the UNIX gid that the new process should setgid() to after * fork()ing and and before spawning any threads. * @param gids null-ok; a list of UNIX gids that the new process should * setgroups() to after fork and before spawning any threads. * @param debugFlags bit flags that enable debugging features. * @param rlimits null-ok an array of rlimit tuples, with the second * dimension having a length of 3 and representing * (resource, rlim_cur, rlim_max). These are set via the posix * setrlimit(2) call. * * @return 0 if this is the child, pid of the child * if this is the parent, or -1 on error. */ native public static int forkAndSpecialize(int uid, int gid, int[] gids, int debugFlags, int[][] rlimits);
22. A method according to claim 10, further comprising: setting operating system level resource	Android includes methods for performing the step described in this claim. *See* claim 21.

The ’720 Patent	Infringed By
management parameters on the child runtime system process.	

Exhibit F

DONALD F. ZIMMER, JR. (SBN 112279)
fzimmer@kslaw.com
CHERYL A. SABNIS (SBN 224323)
csabnis@kslaw.com
KING & SPALDING LLP
101 Second Street – Suite 2300
San Francisco, CA 94105
Telephone: (415) 318-1200
Facsimile: (415) 318-1300

SCOTT T. WEINGAERTNER (*Pro Hac Vice*)
sweingaertner@kslaw.com
ROBERT F. PERRY
rperry@kslaw.com
BRUCE W. BABER (*Pro Hac Vice*)
bbaber@kslaw.com
KING & SPALDING LLP
1185 Avenue of the Americas
New York, NY 10036-4003
Telephone: (212) 556-2100
Facsimile: (212) 556-2222

IAN C. BALLON (SBN 141819)
ballon@gtlaw.com
HEATHER MEEKER (SBN 172148)
meekerh@gtlaw.com
GREENBERG TRAURIG, LLP
1900 University Avenue
East Palo Alto, CA 94303
Telephone: (650) 328-8500
Facsimile: (650) 328-8508

Attorneys for Defendant
GOOGLE INC.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

ORACLE AMERICA, INC.

Plaintiff,

v.

GOOGLE INC.

Defendant.

Case No. 3:10-cv-03561-WHA

Honorable Judge William Alsup

DEFENDANT GOOGLE INC.'S FOURTH SUPPLEMENTAL RESPONSES TO PLAINTIFF'S INTERROGATORIES, SET ONE, NO. 3

Pursuant to Rule 33 of the Federal Rules of Civil Procedure, Defendant Google Inc. ("Google"), through its attorneys, provides its *Fourth Supplemental Responses to Plaintiff's Interrogatories to Defendant Google Inc., Set One, No. 3*, which interrogatories were served by plaintiff Oracle America, Inc. ("Plaintiff" or "Oracle") on December 2, 2010, as follows.

GENERAL OBJECTIONS (SUPPLEMENTED)

1. Google responds generally that discovery has not yet completed and its investigations of the facts relevant to this litigation are ongoing. Google's responses herein are given without prejudice to Google's right to amend or supplement in accordance with Rule 26(e) of the Federal Rules of Civil Procedure, the Civil Local Rules, the Court's Supplemental Order to Order Setting Initial Case Management Conference, any applicable Standing Orders, and the Case Management Order entered by the Court.

2. Google generally objects to Plaintiff's Interrogatories, and the "Definitions and Instructions" related thereto, to the extent they are inconsistent with or impose obligations beyond those required by the Federal Rules of Civil Procedure, the Civil Local Rules, the Patent Local Rules, the Court's Supplemental Order to Order Setting Initial Case Management Conference, any applicable Standing Orders, and the Case Management Order entered by the Court. In responding to each Interrogatory, Google will respond as required under Rule 33 of the Federal Rules of Civil Procedure.

3. Google objects to Oracle's definition of "Java Platform" on the grounds that the definition is overbroad and misleading to the extent it purports to include "the Java programming language," as to which Oracle does not own proprietary rights. When used in Google's responses, the phrase "Java Platform" shall not include "the Java programming language" and, without acknowledging or agreeing that Oracle owns any proprietary rights in any elements thereof, shall have the meaning ascribed to that phrase in paragraph 9 of Oracle's Amended Complaint, namely "a bundle of related programs, specifications, reference implementations, and developer tools and resources that allow a user to deploy applications written in the Java programming language on servers, desktops, mobile devices, and other devices," including but not limited to the Java compiler, the Java Virtual Machine, the Java Development Kit, the Java

Runtime Environment, the Just-In-Time compiler, Java class libraries, Java application programming interfaces, and Java specifications and reference implementations.

4. Google generally objects to Plaintiff's Interrogatories to the extent (a) they are not reasonably calculated to lead to the discovery of admissible evidence that is relevant to any claim of defense of any party; (b) they are unreasonably cumulative or duplicative; (c) they seek information that is obtainable from some other source that is more convenient, less burdensome, or less expensive; or (d) the burden or expense of the proposed discovery outweighs any likely benefit.

5. Google generally objects to Plaintiff's Interrogatories to the extent they seek information, documents, and/or things protected from discovery by the attorney-client privilege, the work product doctrine, the common-interest privilege, and/or any other applicable privilege, immunity, or protection. Nothing contained in Google's responses is intended to be, or in any way shall be deemed, a waiver of any such applicable privilege or doctrine.

6. Google generally objects to Plaintiff's Interrogatories to the extent they request information, documents, and/or things not within the possession, custody, or control of Google, that are as readily available to Plaintiff as to Google, or that are otherwise in the possession of Plaintiff, on the grounds that such requests are unduly burdensome.

7. Google objects to Interrogatory No. 3 as ambiguous due to the reference to "Google's pleading." This literally reads as a request for Google's bases for its defenses at the time of pleading either its Answer to Oracle's Complaint or Answer to Oracle's Amended Complaint, but does not specify the pleading to which it is referring. Oracle has clarified in writing that it is seeking "Google's factual and legal bases for its defense known to it as of October 4, 2010, November 10, 2010, and now." (January 12, 2011 Letter, Jacobs to Weingaertner.) Google objects that, with this interpretation, Oracle has in effect propounded three separate interrogatories for each of Interrogatories Nos. 3 through 16, for a total of 54 interrogatories, far exceeding the 25 permitted under Rule 33(a) of the Federal Rules. Google further objects that the burden of responding to a request going to its bases for its October 4,

2010 pleading, which has been replaced as the operative pleading in the case, greatly outweighs any benefit.

Notwithstanding the foregoing and the fact that reading each interrogatory as two separate interrogatories still exceeds the limits of Rule 33, Google will respond with respect to when it filed its operative pleading in the case, namely *Google Inc.'s Answer to Plaintiff's Amended Complaint for Patent and Copyright Infringement and Amended Counterclaims* on November 10, 2010 (Doc. #51) ("Answer and Counterclaims"), as well as its bases for its defenses generally, subject to Google's general objection that discovery has just begun, and Google is still developing its defenses. Google reserves the right to object to any additional interrogatories propounded by Oracle due to their already exceeding the limitations, and in view of the Court's admonition that "no enlargements of the limitations on discovery in the Federal Rules of Civil Procedure will be allowed until after counsel have demonstrated that they will behave reasonably in the discovery already authorized."

8. Google further objects to Interrogatory No. 3 due to the use of the phrase "affirmative defense." Google's Answer and Counterclaims does not refer to the defenses as "affirmative defenses," and Google objects to the use of the term to the extent Oracle is attempting to suggest any burden in relation to any of the defenses beyond what is required by any applicable statute or case law.

9. Google further objects to Interrogatory No. 3 for specifically seeking attorney work product and attorney-client privileged information.

10. Google further objects to Interrogatory No. 3 as unduly burdensome and not reasonably calculated to lead to the discovery of admissible evidence. The bases for Google's pleading of its defenses is of little value at this point because its bases for maintaining its allegations have changed since the filing of Google's Answer and Counterclaims due to, for example, the receipt of Oracle's Patent Local Rule 3-1 disclosures and subsequent supplementation, including Oracle's final supplementation received on April 1, 2011. Google's bases may and likely will continue to change as discovery unfolds. Google further notes that courts in this District have expressed skepticism as to the use of contention interrogatories,

particularly early in discovery.

11. Google further objects to Interrogatory No. 3 for implying any pleading obligations on Google beyond those required by the Federal Rules of Civil Procedure, the Civil Local Rules, the Patent Local Rules and any applicable Standing Orders.

12. Google further objects to Interrogatory No. 3 for seeking the bases for pleading in response to Oracle's Amended Complaint, which was deficient in many respects, or for seeking Google's current positions on its defenses in view of Oracle's Patent Local Rule 3-1 disclosures, which remain deficient. (*See* Oracle's Second Supplemental Patent Local Rule 3-1 Disclosure of Asserted Claims and Infringement Contentions served on April 1, 2011 (hereinafter "Final Patent L.R. 3-1 Contentions").) Oracle's Amended Complaint did not identify any asserted claims of the Patents-in-Suit; Oracle's Final Patent L.R. 3-1 Contentions do not identify accused products with any reasonable specificity on a claim by claim basis; and both documents failed to include any factual allegation for many elements as to which Oracle has the burden of proof. Any response below does not constitute a waiver of any work product or attorney-client privileged material relating to Google's interpretation of the numerous ambiguities contained in Oracle's Amended Complaint and Final Patent L.R. 3-1 Contentions. Further, any response below should not be considered any affirmative representation that Oracle has presented a cognizable claim, met its Rule 8 obligations or met its obligations under Patent Local Rule 3-1, or that particular information in the response will represent an applicable basis in the future as further clarity as to Oracle's allegations is attained. Google further objects to each patent-related Interrogatory as unnecessary in view of the specific disclosures contemplated by the Patent Local Rules as well as premature at least because claim terms have not been finally construed.

13. Oracle supplemented its Patent Local Rule 3-1 contentions for the final time on April 1, 2011. Remarkably, months after being explicitly put on notice that its contentions were deficient for failing to identify actual use in accused devices, and after multiple supplementations allowed by Google, Oracle has still refused to analyze an actual Accused Instrumentality of "mobile devices running Android" and instead relies on unsupported conclusions based on an emulator. After a meet and confer at the Court on April 6, 2011, Oracle agreed in front of the

Court that "Oracle has declined Google's offer to allow Oracle to supplement its infringement contentions again. Oracle has chosen to rely on its infringement contentions as currently framed." Google's supplemental response to Interrogatory No. 3 is therefore expressly subject to, and without waiver of, Google's position that Oracle's disclosures and contentions are deficient and is made to the best of Google's ability based on Oracle's contentions as Google presently understands them. Google has withdrawn its offer to consent to an amendment of the contentions and reserves the right to move to strike or dismiss claims due to the deficiencies in Oracle's Final Patent L.R. 3-1 Contentions.

14. Despite the fact that Oracle is seeking to combine multiple distinct interrogatories into one in Interrogatories No. 3 through 16 and thereby exceeding its permitted number of interrogatories, Google provides the following responses with respect to its bases for both pleading its defenses in its Amended Answer and Counterclaims, as well as its bases for maintaining them. Google explicitly preserves its work product and attorney-client privileged information and other relevant objections. Google has conducted a reasonable inquiry sufficient to comply with any obligations with respect to these Interrogatories, and makes no representation that these responses include an exhaustive list of all facts relevant to the defenses identified in these Interrogatories. Inclusion of Oracle's allegations in a list of facts in any response herein does not mean that Google agrees with the veracity of the allegation, but merely references the fact that particular allegations were made. Google expressly maintains all objections made in responsive pleadings. Google makes no representation that its responses below completely set forth all of its bases for its defenses, as Google objects that such a response would be unduly burdensome, premature, and require the unwarranted disclosures of attorney work product and attorney-client privileged information.

15. Google incorporates by reference these General Objections into the specific objections and responses set forth below. While Google may repeat a General Objection for emphasis or some other reason, the failure to specifically refer to any General Objection does not constitute a waiver of any sort. Moreover, subject to the requirements of Rule 33 of the Federal

Rules, Google reserves the right to alter or amend its objections and responses set forth herein as additional facts are ascertained and analyzed.

16. Google remains willing to meet and confer with respect to any of its objections to assist Plaintiff in clarifying or narrowing the scope of the requested discovery, and reserves the right to move for a protective order if agreement cannot be reached.

SPECIFIC OBJECTIONS AND RESPONSES

Google's responses to Plaintiff's Interrogatories are based upon Google's current information and belief as a result of reasonable searches and inquiries. Google reserves its right to amend and supplement its responses as it learns additional facts.

INTERROGATORY NO. 3:

Please explain the factual and legal bases for Google's pleading of its first affirmative defense: No Patent Infringement.

FOURTH SUPPLEMENTAL RESPONSE:

In addition to its General Objections, Google objects to this Interrogatory as it seeks information protected by the attorney-client privilege, the work product doctrine, and/or any other applicable privilege, immunity, or protection. Google further objects to this Interrogatory as unduly burdensome as it is not reasonably calculated to lead to the discovery of admissible information. Google further objects to the request to "explain" factual bases as vague and ambiguous. Google further objects to any implication in this Interrogatory that Google has any burden beyond what is required by any applicable statute or case law. Google further objects that Oracle has not complied with its Patent Local Rule 3-1 obligations and Oracle's Final Patent L.R. 3-1 Contentions remain unclear and incomplete. Oracle has also included new conclusory statements regarding how the Android Compatibility Test Suite and emulator purportedly support Oracle's contentions. Although still plainly deficient, as discussed below, Oracle should have provided this information at the outset rather than four months after it initially served its contentions. Further, Oracle did not provide any of the underlying evidence and instead included

only conclusions with no facts or evidentiary support. After prompting, on April 26, 2011, Oracle provided "lists of Android routines that, in the manner described in Oracle's infringement contentions, we observed in operation." Notably this response provides a single set of two lists without specificity as to a particular release of Android, despite Oracle being on notice that there are very significant differences in certain releases, further demonstrating Oracle's lack of support for its contentions. Google will analyze this information and supplement accordingly.

Because Oracle has now stated on the record that it considers these contentions final and refused additional supplementation, Google will endeavor to provide complete responses to the best of its ability in view of the lack of clarity provided by Oracle. This analysis will take longer than the short period of time provided for this supplementation pursuant to the agreement and Google will supplement again as it continues to conduct its analysis of Oracle's Final Patent L.R. 3-1 Contentions.

In view of Oracle's insistence for supplementation of this interrogatory response prior to the Court's claim construction ruling, Google provides this response subject to the caveat that its positions may change significantly with Google's continued analysis and development of its claim construction positions through the processes contemplated by Patent Local Rules and this Court's schedule.

Inclusion of Oracle's allegations in the list of facts in this response does not mean that Google agrees with the veracity of the allegation, but merely references the fact that particular allegations were made. Google expressly maintains all objections made in responsive pleadings. Google further objects to this Interrogatory as unnecessary in view of the specific disclosures contemplated by the Patent Local Rules as well as premature at least because no claim terms have been finally construed and any response herein is made in view of the lack of certainty with respect to the resolution of the meaning of claim terms.

Subject to the foregoing objections and the General Objections, without waiver or limitation thereof, Google states that the following facts relevant to this defense were in its possession or accessible to Google at the time it pleaded this defense in its Answer and Counterclaims:

- Allegations contained in Oracle's Complaint and Exhibits (Doc. #1).
- Allegations contained in Oracle's Amended Complaint and Exhibits (Doc. #36).
- Allegations contained in presentation materials received from Oracle pursuant to Fed. R. Evid. 408.
- The patents-in-suit and their prosecution histories.
- Android source code.
- Android documentation, including public documentation located at http://source.android.com/; http://developer.android.com/; http://code.google.com/android/; http://sites.google.com/site/io/dalvik-vm-internals.

The information cited above, coupled with the positions stated in Google's Amended Answer and Counterclaims, provides Oracle with fair notice of the bases for Google's defenses at the time of the pleading in view of the fact that at the time, Oracle had not disclosed any specific theory of infringement or identified a single asserted claim. Google objects to any further explication of it bases at the time of the pleading as unduly burdensome.

General Allegations in Oracle's Final Patent L.R. 3-1 Contentions

As an initial matter, Oracle continues to assert over 130 claims, making this analysis time consuming and inefficient. This stands in stark contrast to its continued demands for complete supplementation and purported intent to proceed efficiently. Also, as noted by the Court, Oracle will have to reduce its claims to one or two claims and yet so far has not dropped a single claim.

With respect to Oracle's identification of Accused Instrumentalities, Oracle does not provide the requisite specificity, instead merely identifying "Android," the "Android Platform," "mobile devices running Android," and providing "representative examples" of devices running Android. (*See* Section "Patent Local Rule 3-1(b) – Accused Instrumentalities", Oracle's Final Patent L.R. 3-1 Contentions at p. 2.) This does not provide fair notice of each of the purported Accused Instrumentalities and further, certain of the Exhibits, such as Exhibit F, identify, for example, "computers running the Android SDK," which contradicts the list of Accused Instrumentalities on page 2 of its cover document. Oracle also identifies "Google Dev Phones, the Google Nexus One, and the Google Nexus S" as purported "Google devices running

Android." For one, Oracle did not analyze a single one of these devices. Further, Oracle's support for the fact that the Samsung Nexus S is purportedly a Google device is a link from a web site speculating that the Nexus S is the successor to the Nexus One. Oracle ignores the fact that the device is manufactured by Samsung. Oracle also added "the Google Dev Phones" for the first time in its last supplementation without providing any further explanation as to what these devices are or what basis it has for it allegations. Therefore, each of these allegations is deficient on its face. Similarly, the allegations for the "other mobile devices" are deficient for the reasons stated herein.

In its Final Patent L.R. 3-1 Contentions Oracle included an allegation of inducement pursuant to 35 U.S.C. § 271(b) because Google purportedly "contracts with, instructs, and otherwise induces others. . . ." (*Id.* at 3.) In Section D, "Indirect Infringement," the entirety of the purported inducing acts are described in vague generalities:

> On information and belief, Google purposely and actively distributes the Accused Instrumentalities to manufacturers of products and application developers with the intention that they be used, copied, and distributed to consumers, who in turn use them. Google induces and contributes to the infringement of the asserted claims of each asserted patent, because Google encourages manufacturers, application developers, and service providers (including the members of the Open Handset Alliance), as well as end users, to copy, sell, distribute, re-distribute, and use products that embody or incorporate the Accused Instrumentalities.

(*Id.* at 7.) This section does not even distinguish between purported inducement and purported contributory infringement. It also fails to identify inducing acts specific to each purported Accused Instrumentality as Oracle identifies them in Section B. Oracle does not even allege that Google "distributes," the Accused Instrumentalities identified as "other mobile devices running Android" making the first sentence of the block quote plainly incorrect. Further, Oracle fails to connect any purportedly inducing act with the purported direct infringers. For example, to the extent the user of "other mobile devices running Android" is purportedly a direct infringer, Oracle cannot maintain an inducement allegation against Google with a third party between Google and the user (Oracle's own allegation states only that Google distributes to "manufacturers of products" and "application developers"). This provision thus fails to provide the requisite specificity of inducing acts for each type of Accused Instrumentality and without an

inducing act, Oracle cannot prevail on a claim pursuant to 35 U.S.C. § 271(b). Further, Oracle has not made an allegation or provided any evidence that Google had the requisite scienter required for a claim pursuant to 35 U.S.C. § 271(b). Oracle alleges only that Google purportedly "purposely and actively distributes the Accused Instrumentalities to manufacturers of products and application developers with the intention that they be used . . . ." (*Id.* at 7.) Notably this statement at best suggests a purported intent that certain instrumentalities be used, not an intent to infringe. Further, Oracle includes a single statement that Google "has actual knowledge of Oracle's patents" (*see id.* at 8) at the end of the indirect infringement allegation, but this statement is deficient at least because it does not even allege Google "had" knowledge and specific intent required for inducement. Further, the purported "evidence" of knowledge, such as imputing the knowledge of inventors on Google is facially deficient to establish specific intent.

Oracle also includes an allegation of contributory infringement pursuant to 35 U.S.C. § 271(c). Oracle alleges only that Google purportedly "offers to sell, sells, or imports part or all of the Accused Instrumentalities within or into the United States." (*Id.* at 8.) This statement provides no specificity as to the component that Google is accused of offering to sell, which component Google is accused of selling, or which component Google is accused of importing. The statement similarly fails to identify with specificity if each allegation goes to "part" or "all" of a particular Accused Instrumentality. Oracle cannot prevail on a claim pursuant to 35 U.S.C. § 271(c) for any Accused Instrumentalities that are downloadable source files because they are not "components" within the meaning of 35 U.S.C. § 271(c). Oracle cannot prevail on a claim pursuant to 35 U.S.C. § 271(c) for any Accused Instrumentalities that are made available for download because they are not sold or offered for sale within the meaning of 35 U.S.C. § 271(c).

Oracle has not demonstrated that the Accused Instrumentalities are not capable of substantial non-infringing uses. Notably, Oracle provides a sole example of a discussion of the "dx tool from the Android SDK," in the context of substantial non-infringing uses, but the dx tool is not even listed as an Accused Instrumentality in the preceding paragraph. For one, this statement focuses on the dx tool even though the Android SDK is identified as the Accused

Instrumentality. Further, setting aside Google's position that all uses of the dx tool are non-infringing, Oracle's contentions are still factually incorrect as the dx tool has a myriad of substantial non-infringing uses, which are not even purported to be infringing by Oracle. The dx tool can translate Dalvik assembly language into Dalvik bytecode; it contains a .class file dumper/disassembler; it can perform partial translations of .class files; it can generate a human-readable dump of .dex files; and it provides a code annotation mode called "annotool." This is merely an exemplary list and an example of how, if Oracle made an effort to meet its burden on this issue, Google could provide a response. Additional examples of substantial non-infringing uses were cited Google's Answer to Oracle's Amended Complaint (Dkt. 51) at ¶ 18 in that various programming languages that can be used to create software applications. Additionally, the Android NDK (http://developer.android.com/sdk/ndk/index.html) allows Android developers to implement applications or parts of applications using native-code languages such as C and C++. This does not use the Java programming language and therefore constitutes a substantial non-infringing use of the Android platform as a whole. Finally, as presently understood, all uses are noninfringing uses for the reasons cited herein.

Oracle also includes an allegation of a purported violation of 35 U.S.C. § 271(f) because Google purportedly "supplies part or all of the Accused Instrumentalities in or from the United States to foreign contractors, including HTC. . . ." (Final Patent L.R. 3-1 Contentions at 3.) Oracle fails to identify with specificity which Accused Instrumentalities it is referring to and if each allegation goes to "part" or "all" of a particular Accused Instrumentality. Oracle cannot prevail on a claim pursuant to 35 U.S.C. § 271(f) for any Accused Instrumentalities that are downloadable source files because they are not "components" within the meaning of 35 U.S.C. § 271(f) . *See Microsoft Corp. v. AT&T Corp.*, 550 U.S. 437 (2007).

Oracle also fails to provide evidence of purported infringement by devices such as third party devices, instead relying on conjecture and conclusion as to their operation as opposed to investigation. Oracle's citation in Footnote 10 of its Final Patent L.R. 3-1 Contentions contradicts Oracle's own assumptions that implementers cannot modify Android code, as the cited language states that "Device implementers MAY modify the underlying implementation of

the APIs" As discussed below, Oracle relies on voluntary application level compatibility using the Android compatibility test suite (CTS) to draw incorrect and unsupported inferences about lower level functionality that Google specifically states can be modified even by third parties seeking compatibility approval. Oracle also fails to provide evidence for each of the versions of Android that Oracle purports to accuse and instead relies on representative evidence, despite having access to multiple versions that are freely available to the public via the public git repository. Perhaps the most notable flaw, however, is that Oracle made a specific, calculated decision to forego examination of actual devices. Although the CTS can be run against an emulator, as Oracle did, it is designed to test actual devices. (*See* http://source.android.com/compatibility/cts-intro.html ("Attach at least one device (or emulator) to your machine.").) Yet Oracle inexplicably chose not to run the CTS against even a single one of the exemplary devices identified in its Final Patent L.R. 3-1 Contentions as Accused Instrumentalities and instead continues to rely on an unsupported assumption as to behavior of actual devices.

Oracle's statement that "[t]o the extent that any element or limitation of the asserted claims is not found to have literal correspondence in the Accused Instrumentalities, Oracle alleges, on information and belief, that any such elements or limitations are present under the doctrine of equivalents in the Accused Instrumentalities" is deficient as a matter of law and, as these are Oracle's Final Patent L.R. 3-1 Contentions, Oracle is barred from raising doctrine of equivalents arguments in the future. *See Optimumpath LLC v. Belkin International Inc., et. al.*, 4-09-cv-01398 (N.D. Cal., April 12, 2011, Order) (Wilken, J.) (catch-all doctrine of equivalents allegation fails to comply with Patent Local Rules).

Oracle's Final Patent L.R. 3-1 Contentions contains an irrelevant contention purporting that "Android [is] not an open platform." As a threshold matter, this statement is plainly irrelevant and immaterial to the infringement contentions as whether a platform is "open" or not has no bearing on this inquiry. Further, the statement is misleading as it cites purported "recent actions," but references an "anti-fragmentation program" that has been in place for four years - since 2007. Oracle further attempts to mischaracterize the delay in the release of the open source

of Honeycomb - a version of Android for tablet devices that has not been accused in the current litigation. This is a temporary delay while the Android team works to bring the Honeycomb features to mobile phones, after which the open source will be made available, consistent with Android's open strategy. Google remains firmly committed to providing Android as an open source platform across many device types.

Further, with respect to Google's other bases for maintaining its defense that Oracle cannot establish infringement of any asserted claim, Google states the following based on Oracle's contentions as presently understood, with reference to Oracle's Final Patent L.R. 3-1 Contentions:

The '104 Reissue Patent

- **General:** For direct infringement Oracle accuses "device[s] running Android," for apparatus and system claims, "storage devices containing Android code," for "computer program product, memory, and computer-readable medium claims," and "[a]nyone who uses a device running Android code" for method claims. (*See* Ex. A to Oracle's Final Patent L.R. 3-1 Contentions at 2.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an actual performance of the method. Instead, Oracle states that 1) "Oracle has determined that Android devices execute much of the code cited below every time the devices start up" and 2) "Oracle determined that many of these code portions are executed even before a user can interact with a device" based on analysis using the Android Compatibility Test Suite (CTS) and a "mobile device emulator" included with the Android SDK. (*See* Ex. A at 1-2.) Oracle does not provide any of the factual evidence of this analysis and the analysis does not support the conclusion that Oracle draws regarding the operation of actual devices. First, even if true, executing "much" of the code, and "many of the[] code portions," cannot demonstrate infringement particularly when Oracle does not provide any break down of what code or code portions it cites in its claim charts are admittedly not executed based in its own analysis. Second, although a trace log of the emulator may show that certain methods are called, it does not mean the CTS checks for the existence of these methods, and, as such, a device could pass compatibility

testing with modifications that would result in those methods not being called. Google is continuing to perform its own analysis, which is hampered by Oracle's failure to timely provide any factual basis and continued failure to provide the complete factual basis for its contentions and Google will supplement these responses as necessary.

- **Claims 11, 19, 21, 22, 25, 28, 31, 34, 37, 40, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit A any specific device that allegedly infringes, and has instead relied on general statements referring to unidentified "devices that run Android" or "devices that run Android and the Android SDK." The latter statement is inconsistent with the recitation of Accused Instrumentalities in the top of the Exhibit A, which does not include Android SDK. (*See* Ex. A at 1-2.) Oracle has not made a showing of infringement because it has not identified any specific alleged infringing device or purported direct infringer for these claims, and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of any actual device is flawed.
- **Claims 12, 13, 17-18, 20, 23-24, 26, 27, 29-30, 32-33, 35-36, 38-39, and 41, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit A the actual performance of any allegedly infringing method, and has instead relied on general statements such as "Android includes methods . . .," "devices that run Android and the Android SDK," or "devices that store, distribute, or run Android or the Android SDK, including websites, servers, and mobile devices." The statements regarding Android SDK are inconsistent with the recitation of Accused Instrumentalities at the top of the Exhibit A, which does not include Android SDK. (*See* Ex. A at 1-2.) All of these claims implicate the performance of a method and the charts in Exhibit A are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because it has not identified any alleged infringing act or purported direct infringer for these claims and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.

- **All asserted claims:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for element 11-b, "a processor configured to execute said instructions containing one or more symbolic references by determining a numerical reference corresponding to said symbolic reference, storing said numerical references, and obtaining data in accordance to said numerical references," on pages 12-65 of Exhibit A does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because (1) it would not employ a numerical reference as recited in that it would not use or store a direct reference to the data itself; and/or (2) it would not employ a symbolic reference as recited in that it would not use a non-numeric reference to the data. Each other independent claim in Exhibit A references a similar citation for a similar element and the same basis applies to each of those claims. (*See, e.g.,* Claim 12 ("*See* Claim 11-b, *supra*" in chart for "resolving a symbolic reference . . ." element).) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.
- **All asserted claims:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for element 11-b, "a processor configured to execute said instructions containing one or more symbolic references by determining a numerical reference corresponding to said symbolic reference, storing said numerical references, and obtaining data in accordance to said numerical references," on pages 12-65 of Exhibit A does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not perform the "determining," "storing," and "obtaining" steps as recited in that it would not perform these steps during execution of the instructions. Each other independent claim in Exhibit A references a similar citation for a similar element and the same basis applies to each of those claims. For

example, Claim 12 recites "interpreting said instructions" and "resolving a symbolic reference in an instruction being interpreted." Accordingly, and as described above, the material cited in Exhibit A does not perform the "resolving" step as recited in that it would not perform this step while the instruction is being interpreted. While the other independent claims may employ different language, they require the resolving step to occur while an instruction is being executed and/or interpreted. (*See, e.g.,* Claim 13 ("[a] computer-implemented method for executing instructions . . ."), Claim 17 ("a method for executing said program . . ."), Claim 19 ("a memory for use in executing a program . . ."), Claim 21 ("such that when the program is executed by the processor each symbolic field reference is resolved . . ."), Claim 24 ("determining immediately prior to execution whether a bytecode of the program contains a symbolic reference").) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example, and that Oracle's allegations in its Final Patent L.R. 3-1 Contentions support Google's reasoning. More specifically, Oracle states that 1) "Oracle has determined that Android devices execute much of the code cited below every time the devices start up" and 2) "Oracle determined that many of these code portions are executed even before a user can interact with a device." (*See* Ex. A at 1-2.) These statements evidence that the steps recited in these claims, if performed at all, are not performed during execution of the instructions, but at some other time, such as at boot time. Regardless, Google's investigation into Oracle's recent allegations related to the emulator is ongoing, and Google will supplement as it continues its investigation.

- **All asserted claims:** As presently understood, Oracle has not made a showing of infringement at least because the material cited in Exhibit A relates to a hybrid compiler-interpreter, which is not within the scope of the asserted claims because the patentee explicitly disclaimed such claim scope during prosecution of the asserted claims.

- **Claims 11, 22, and 25:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for element 11-b, "a processor configured to execute said instructions containing one or more symbolic references by determining a numerical reference corresponding to said symbolic reference, storing said numerical references, and obtaining data in accordance to said numerical references," on pages 12-65 of Exhibit A does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not employ a processor configured to execute an instruction as recited in that the listed devices do not have processors for executing intermediate form object code. Independent claims 22 and 25 in Exhibit A reference a similar citation for a similar element and the same basis applies to each of those claims. (*See, e.g.,* Claim 22 ("*See* Claim 11-b, *supra*" in chart for "a processor configured to execute the instruction . . ." element); Claim 25 ("*See* Claim 11, *supra*" in chart for "instructions for causing the processor to . . . execute thereafter the bytecode . . ." element).) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.
- **Claims 12-21, 23-32, and 36-41:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for the "resolving a symbolic reference in an instruction being interpreted, said step of resolving said symbolic reference including the substeps of" element on page 69 of Exhibit A—which is simply a reference to the materials cited for element 11-b at pages 12-65—does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not employ resolving in that it would not rewrite the instruction with the numeric reference. Each other independent claim listed references a similar citation for similar claim elements and the same basis applies to each of those claims. (*See, e.g.,* Claim 13 "*See* Claim 11-b,

supra" in chart for "resolving a symbolic reference . . ." element).) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **Claims 24-26, 30-32, 36-38:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for the "when it is determined that the bytecode of the program contains a symbolic data reference, invoking a dynamic field reference routine to resolve the symbolic data reference" element on page 79 of Exhibit A—which is simply a reference to the materials cited for element 11-b at pages 12-65—does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not employ a dynamic field reference routine in that it would not (1) rewrite the symbolic reference in the in bytecode with a numeric reference, and/or (2) execute the bytecode with the numeric reference prior to advancing to the next instruction. Each other independent claim listed references a similar citation for similar claim elements and the same basis applies to each of those claims. (*See, e.g.,* Claim 25 ("*See* Claim 11, *supra*" in chart for "a memory comprising a program . . ." element).) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.
- **Claims 27-29**: As presently understood, Oracle has not made a showing of infringement at least because the material cited for the "generating a set of new instructions for the program that contain numeric references resulting from invocation of a routine to resolve any

symbolic data references in the set of original instructions" element on page 83 of Exhibit A—which is simply a reference to the materials cited for element 11-b at pages 12-65—does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not employ generating a set of new instructions that contain numeric references as recited. Each other independent claim listed references a similar citation for similar claim elements and the same basis applies to each of those claims. (*See, e.g.,* Claim 28 ("*See* Claim 11, *supra*" in chart for "a memory comprising a control program . . ." element).) In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **Claims 30-32:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for the "replacing each instruction in the program with a symbolic data reference with a new instruction containing a numeric reference resulting from invocation of a dynamic field reference routine to resolve the symbolic data reference" element on page 85 of Exhibit A—which is simply a reference to the materials cited for Claim 11 at pages 1-65—does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit A because it would not replace instructions in the program as recited. Each other independent claim listed references a similar citation for similar claim elements and the same basis applies to each of those claims. (*See*., e.g., Claim 31 ("*See* Claim 11, *supra*" in chart for "a memory comprising a control program for causing the processor to . . . replace each instruction in the program with a symbolic data reference . . ." element).) In particular, the identified DEX optimization does not replace any instructions at run-time in the matter required by the claims, to the extent that any replacement of instructions may occur. In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the

preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. Further, uses such as the Android NDK are substantial non-infringing uses as they do not implicate the accused subject matter cited in Oracle's charts. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.
- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

The '205 Patent

- **General:** For direct infringement Oracle accuses "device[s] running Android," for apparatus claims, and "[a]nyone who uses a device running Android" for method claims. (*See* Ex. B-1 to Oracle's Final Patent L.R. 3-1 Contentions at 1; Ex. B-2 at 2.)
- **All Asserted Claims:** For these claims, Oracle has not provided evidence of actual operation on an actual device or evidence of an actual performance of the method. Instead, Oracle states that 1) "versions of Android . . . have the Jit.c code activated by default" (Ex. B-1 at 1), or 2) "Oracle has determined that Android devices execute much of the code cited below every time the devices start up" and 3) "Oracle determined that many of these code portions are executed even before a user can interact with a device" (Ex. B-2 at 1-2), the latter two statements are based on analysis using the Android Compatibility Test Suite (CTS) and a "mobile device emulator" included with the Android SDK. Oracle does not provide

any of the factual evidence of this analysis and the analysis does not support the conclusion that Oracle draws regarding the operation of actual devices. With respect to Jit.c, whether or not it is activated by default does not establish how third parties use it and does not amount to proof of the use in any device. With respect to Ex. B-2, even if true, executing "much" of the code, and "many of the[] code portions," cannot demonstrate infringement particularly when Oracle does not provide any break down of what code or code portions it cites in its claim charts are admittedly not executed based in its own analysis. Second, although a trace log of the emulator may show that certain methods are called, it does not mean the CTS checks for the existence of these methods, and, as such, a device could pass compatibility testing with modifications that would result in those methods not being called. Google is continuing to perform its own analysis, which is hampered by Oracle's failure to timely provide any factual basis and continued failure to provide the complete factual basis for its contentions and Google will supplement these responses as necessary.

- **Claims 1-4:** As presently understood, Oracle has not made a showing of infringement, even if the materials cited in Exhibits B-1 or B-2 were implemented and used in a device in the form recited at least because:
    - It would not employ a method in a computer system where a virtual machine instruction was generated at runtime because a virtual machine instruction ("[a] machine instruction for a software emulated microprocessor or computer architecture (also called virtual code)") is never generated in the functionality identified in Ex. B-1, and the functionality identified in Ex. B-2 is not implemented at runtime.
    - It would not employ a method in a computer system where a virtual machine instruction represents or references one or more native instructions because no new virtual machine instruction represents or references one or more native instructions ("[a] machine instruction that is designed for a specific microprocessor or computer architecture (also called native code)"). (*See* Exs. B-1 and B-2.)

- It would not employ a method in a computer system where a new virtual machine instruction was executed instead of a first virtual machine instruction, because a new virtual machine instruction is not generated in the functionality identified in Ex. B-1 and a new virtual machine instruction is not executed in the functionality identified in Ex. B-2.

In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **Claim 8:** As presently understood, Oracle has not made a showing of infringement, even if the materials cited in Exhibits B-1 or B-2 were implemented and used in a device in the form recited at least because:
  - Claim 8 includes claim elements similar to those in claim 1 and, to that extent, the same bases cited above apply to this claim.
  - It would not employ a method in a computer system for compiling a portion of the function into at least one native machine instruction so that the function includes both virtual and native machine instruction because no portion of a function is compiled into a native machine instruction. (*See* Ex. B-2.)
  - It would not employ a method in a computer system where at least one native machine instruction with a new virtual machine instruction is executed after the compiling of the function because a native machine instruction is not represented by a new virtual machine code. (*See* Ex. B-1.)

  In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator

example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. Further, uses such as the Android NDK are substantial non-infringing uses as they do not implicate the accused subject matter cited in Oracle's charts. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.
- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

The '702 Patent

- **General:** For direct infringement Oracle accuses "computers running the Android SDK" for apparatus claims, "storage devices containing the Android SDK," for computer-readable medium claims, and "anyone who uses the Android SDK" for method claims. (*See* Ex. C to Oracle's Final Patent L.R. 3-1 Contentions at 1.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an actual performance of the method. Instead, Oracle states that 1) "[D]evelopers must run the Android dx tool to build Android applications, and generate Android bytecode and .dex files, and run the Dalvik virtual machine to test them" and 2) "The Android SDK is a tool used purely to build and test Android programs." (*See* Ex. C at 1.) In its Final Patent L.R. 3-1 Contentions Oracle provides only an illustrative example in its cover document of the dx tool as purportedly having no substantial non-infringing uses. For one, this statement focuses on the dx tool even though the Android SDK is identified as the Accused Instrumentality. Further, setting aside Google's position that all uses of the dx tool are non-infringing,

Oracle's contentions are still factually incorrect as the dx tool has a myriad of substantial non-infringing uses, which are not even purported to be infringing by Oracle. The dx tool can translate Dalvik assembly language into Dalvik bytecode; it contains a .class file dumper/disassembler; it can perform partial translations of .class files; it can generate a human-readable dump of.dex files; and it provides a code annotation mode called "annotool." This is merely an exemplary list and an example of how, if Oracle made an effort to meet its burden on this issue, Google could provide a response. Oracle has changed its contention from being limited to the dx tool to being targeted to the "Android SDK including the dx tool." Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.

- **Claims 1 and 7, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit C the actual performance of any allegedly infringing method and instead relied on a general statement including "Android dx tool involves a method" or "Android dx tool [performs steps]." All of these claims implicate the performance of a method and the charts in Exhibit C are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because Oracle's bare assertion that "developers must run the Android dx tool to build Android applications," (*see* Ex. C at 1) does not amount to proof of an actual use that the dx tool and associated code shown in the charts has been executed in the manner alleged.
- **Claims 13, and all dependent claims that depend therefrom:** For these claims, Oracle failed to identify on a claim by claim basis in Exhibit C any specific device that allegedly infringes and instead relied on a general statement including "[a]ny device or computer which can run the Android dx tool." Oracle has not made a showing of infringement because Oracle's bare assertion that "developers must run the Android dx tool to build Android applications," (*see* Ex. C at 1) does not amount to proof of an actual use that the dx tool and associated code shown in the charts has been executed in a device in the manner alleged.

- **All Asserted Claims:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for the "removing said duplicated elements from said plurality of class files to obtain a plurality of reduced class files" element on pages 13-17 of Exhibit C does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit C because it would not employ a method of obtaining a plurality of reduced class files in that there would be no intermediate step of removing duplicated elements from class files to obtain a plurality of reduced class files prior to forming a multi-class file. Similarly, Oracle has not made a showing of infringement at least because the material cited for the "forming a multi-class file comprising said plurality of reduced class files" element at pages 17-20 of Exhibit C does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit C because it would not employ a method of forming a multi-class file in that no multi-class file would be formed from reduced class files obtained prior to forming the multi-class file. Each other independent claim in Exhibit C references Oracle's citation for claim 1 for similar elements and the same basis applies to those claims. Oracle has changed is contention from being limited to the dx tool to being targeted to the "Android SDK including the dx tool." Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.
- **All Asserted Claims:** As presently understood, Oracle has not made a showing of infringement at least because the material cited for "determining plurality of duplicated elements in a plurality of class files" elements on pages 2–9 of Exhibit C does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit C because it would not employ a method of determining a plurality of duplicated elements in a plurality of class files in that the classes cited do not determine whether a duplicated element is duplicated within a single class file or across two class files or whether the duplicated is one of many or the only one. Each other independent claim in Exhibit C references Oracle's citation for claim 1 for similar elements and the same basis applies to those claims. Oracle has changed is contention from being limited to the dx tool to being

targeted to the "Android SDK including the dx tool." Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.

- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. The dx tool has substantial non-infringing uses as discussed above. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.
- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.
- **The '447 PatentGeneral:** For direct infringement Oracle accuses "device[s] running Android," for system claims, "storage devices containing Android code," for "computer-readable medium claims," and "[a]nyone who uses a device running Android code" for method claims. (*See* Ex. D to Oracle's Final Patent L.R. 3-1 Contentions at 2.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an actual performance of the method. Instead, Oracle states that 1) "Oracle has determined that Android devices execute much of the code cited below when a developer runs the Android Compatibility Test Suite (CTS)" and 2) "Oracle determined that many of these code sections are executed as part of Google's CTS testing." (*See* Ex. D at 1-2.) Oracle does not provide any of the factual evidence of this analysis and the analysis does not support the conclusion that Oracle draws regarding the operation of actual devices. First, even if true, executing "much" of the code, and "many of the[] code sections," cannot demonstrate infringement particularly when Oracle does not provide any break down of what code or code portions it cites in its claim charts are admittedly not executed based in its own analysis. Further, Oracle appears to now be limiting its allegation

to testing only and not alleging that any third party device runs a program that uses this functionality outside of the testing context. Google is continuing to perform its own analysis, which is hampered by Oracle's failure to timely provide any factual basis and continued failure to provide the complete factual basis for its contentions and Google will supplement these responses as necessary. As an initial matter, however, Oracle's statements are plainly false with respect to the Gingerbread release. Further, to the extent the CTS checks for the existence of certain files, that in and of itself does not amount to executing the functionality.

- **Claims 1, 7, 10, and 16, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit D the actual performance of any allegedly infringing method and has instead relied on general statements such as "Android includes methods . . .," or "devices that store, distribute, or run Android or the Android SDK, including websites, servers, and mobile devices." All of these claims implicate the performance of a method and the charts in Exhibit D are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because it has not identified any alleged infringing act or purported direct infringer for these claims and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.
- **Claims 10, and 16, and all dependent claims that depend therefrom:** These claims recite the practice of steps in a closed rather than open manner *i.e.*, without the benefit of inclusive phrasing such as "comprising" in connection with the recitation of the steps. The material cited by Oracle, even if it were implemented and used in a device in the form it is recited in Exhibit D, would include steps not contained in the recited methods, thereby precluding infringement of these claims. In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.

- **Claims 19, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit D any specific device that allegedly infringes and has instead relied on general statements referring to "devices that run Android or the Android SDK." Oracle has not made a showing of infringement because it has not identified any specific alleged infringing device or purported direct infringer for these claims, and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation on a device is flawed.
- **All Asserted Claims:** Oracle has not made, and cannot make, a showing of infringement of any asserted claim. Oracle has not identified any evidence of direct infringement of this patent, *i.e.*, any evidence that any user performs all of the elements of any valid and enforceable claim. For example, Oracle's Infringement Contentions for this patent identifies the class java.lang.SecurityManager as a requisite part of the allegedly infringing behavior. (*See, e.g.,* Exhibit D at p. 12.) However, Oracle has not identified any user or application program that uses SecurityManager. Google does not use SecurityManager, and is not aware of any third party that uses or has ever used SecurityManager. Google further states that, prior to the commencement of this litigation, Google had already implemented changes to disable SecurityManager-related functionality for the Gingerbread version of Android (version 2.3), which was released on December 6, 2010. In the current version of Android, as of December 6, 2010, users and application programs are prevented from even installing an instance of SecurityManager. *See* java.lang.System.setSecurityManager().
- **All Asserted Claims:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit D how each claim is even capable of infringing in the absence of further programming and/or configuration. All of these claims implicate the existence of "protection domains" and "permissions" and the charts in Exhibit D, which only point to class definitions, are devoid of any example of code written to create protection domains or permissions, thereby precluding a finding of infringement. At most, the material referenced in Exhibit D may provide a mechanism that would allow a programmer to write code that

may satisfy one or more elements of these claims, but such a finding would not provide evidence of infringement even under a capable of infringing analysis.

- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.

Oracle's position is especially implausible given that Google does not use the identified SecurityManager, and is not aware of any third party that uses or has ever used SecurityManager, and prior to the commencement of this litigation, Google had already implemented changes to disable SecurityManager-related functionality for the Gingerbread version of Android (version 2.3), which was released on December 6, 2010 and that in the current version of Android, as of December 6, 2010, users and application programs are prevented from even installing an instance of SecurityManager. *See* java.lang.System.setSecurityManager(). Rather than attempt to demonstrate that this accused functionality is actually used, or indeed necessarily used, in Accused Instrumentalities, Oracle has instead represented that its position is that "the statement that 'Google does not use SecurityManager' is not true." (March 13, 2011 Letter, Peters to Weingaertner.) The letter lacks any explanation of how Google (or anyone else for that matter) allegedly uses SecurityManager. As a result, Oracle cannot establish infringement as a matter of law. Indeed, it appears Oracle has acquiesced and is no longer alleging that SecurityManager is used as intended in any non-testing scenario and has changed it allegations to be limited to "when a developer runs the Android Compatability Test Suite" and "as part of Google's CTS testing," (*see* Ex. D at 1-2) although its allegations relating to the testing scenario are flawed as well for the reasons stated herein. Google continues to analyze this new allegation and will supplement as it continues its investigation.

- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

The '476 Patent

- **General:** For direct infringement Oracle accuses "device[s] running Android," for system claims, "storage devices containing Android code," for "computer-readable medium claims," and "[a]nyone who uses a device running Android code" for method claims. (*See* Ex. E to Oracle's Final Patent L.R. 3-1 Contentions at 2.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an actual performance of the method. Instead, Oracle states that 1) "Oracle has determined that Android devices execute much of the code cited below when a developer runs the Android Compatibility Test Suite (CTS)" and 2) "Oracle determined that many of these code sections are executed as part of Google's CTS testing." (*See* Ex. E at 1-2.) Oracle does not provide any of the factual evidence of this analysis and the analysis does not support the conclusion that Oracle draws regarding the operation of actual devices. First, even if true, executing "much" of the code, and "many of the[] code sections," cannot demonstrate infringement particularly when Oracle does not provide any break down of what code or code portions it cites in its claim charts are admittedly not executed based in its own analysis. Further, Oracle appears to now be limiting its allegation to testing only and not alleging that any third party device runs a program that uses this functionality outside of the testing context. Google is continuing to perform its own analysis, which is hampered by Oracle's failure to timely provide any factual basis and continued failure to provide the complete factual basis for its contentions and Google will supplement these responses as necessary. As an initial matter, however, Oracle's statements are plainly false with respect to the Gingerbread release. Further, to the extent the CTS checks for the existence of certain files, that in and of itself does not amount to executing the functionality.
- **Claims 1, 5, 6, 10, 14, and 15, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit E the actual

performance of any allegedly infringing method and has instead relied on general statements such as "Android includes methods . . .," or "devices that store, distribute, or run Android or the Android SDK, including websites, servers, and mobile devices." All of these claims implicate the performance of a method and the charts in Exhibit E are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because it has not identified any alleged infringing act or purported direct infringer for these claims and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.

- **Claims 10, 14, and 15, and all dependent claims that depend therefrom:** These claims recite the practice of steps in a closed rather than open manner, *i.e.*, without the benefit of inclusive phrasing such as "comprising" in connection with the recitation of the steps. The material cited by Oracle, even if it were implemented and used in a device in the form it is recited in Exhibit D, would include steps not contained in the recited methods, thereby precluding infringement of these claims. In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.
- **Claims 19, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit E any specific device that allegedly infringes and has instead relied on a reference to claim 1 which contains a general statement that "Android includes methods. . . ." Oracle has not made a showing of infringement because it has not identified any specific alleged infringing device or purported direct infringer for these claims, and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.
- **All Asserted Claims:** Oracle has not made, and cannot make, a showing of infringement of any asserted claim. Oracle has not identified any evidence of direct infringement of this

patent, *i.e.*, any evidence that any user performs all of the elements of any valid and enforceable claim. For example, Oracle's Infringement Contentions for this patent identifies the class java.lang.SecurityManager as a requisite part of the allegedly infringing behavior. *See, e.g.*, Exhibit E at p. 13. However, Oracle has not identified any user or application program that uses SecurityManager. Google does not use SecurityManager, and is not aware of any third party that uses or has ever used SecurityManager. Google further states that, prior to the commencement of this litigation, Google had already implemented changes to disable SecurityManager-related functionality for the Gingerbread version of Android (version 2.3), which was released on December 6, 2010. In the current version of Android, as of December 6, 2010, users and application programs are prevented from even installing an instance of SecurityManager. *See* java.lang.System.setSecurityManager().

- **All Asserted Claims:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit E how each claim is even capable of infringing in the absence of further programming and/or configuration. All of these claims implicate the existence of a "protection domain" and "permissions" and the charts in Exhibit E, which only point to class definitions, are devoid of any example of code written to create a protection domain or permissions, thereby precluding a finding of infringement. At most, the material referenced in Exhibit E may provide a mechanism that would allow a programmer to write code that may satisfy one or more elements of these claims, but such a finding would not provide evidence of infringement even under a capable of infringing analysis.
- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.

Oracle's position is especially implausible given that Google does not use the identified SecurityManager, and is not aware of any third party that uses or has ever used SecurityManager, and prior to the commencement of this litigation, Google had already implemented changes to disable SecurityManager-related functionality for the Gingerbread version of Android (version 2.3), which was released on December 6, 2010 and that in the current version of Android, as of December 6, 2010, users and application programs are prevented from even installing an instance of SecurityManager. *See* java.lang.System.setSecurityManager(). Rather than attempt to demonstrate that this accused functionality is actually used, or indeed necessarily used, in Accused Instrumentalities, Oracle has instead represented that its position is that "the statement that 'Google does not use SecurityManager' is not true." (March 13, 2011 Letter, Peters to Weingaertner.) The letter lacks any explanation of how Google (or anyone else for that matter) allegedly uses SecurityManager. As a result, Oracle cannot establish infringement as a matter of law. Indeed, it appears Oracle has acquiesced and is no longer alleging that SecurityManager is used as intended in any non-testing scenario and has changed it allegations to be limited to "when a developer runs the Android Compatability Test Suite" and "as part of Google's CTS testing," (*see* Ex. E at 1-2) although its allegations relating to the testing scenario are flawed as well for the reasons stated herein. Google continues to analyze this new allegation and will supplement as it continues its investigation.

- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

The '520 Patent

- For direct infringement Oracle accuses "computers running the Android SDK" for apparatus claims, "storage devices containing the Android SDK," for computer-readable medium claims, and "[a]nyone who uses the Android SDK" for method claims. (*See* Ex. F to Oracle's Final Patent L.R. 3-1 Contentions at 1.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an

actual performance of the method. Instead, Oracle states that 1) "[t]he Android code cited below necessarily infringes because developers must run javac and the Android dx tool to build Android applications and run the Dalvik virtual machine to test them," 2) "the bytecode execution simulator in the dxtool will always process each <clinit> method, including those that initialize static arrays" and 3) "[t]he Android SDK is a tool used purely to build and test Android programs." (*See* Ex. F at 1.) In its Final Patent L.R. 3-1 Contentions Oracle provides only an illustrative example in its cover document of the dx tool as purportedly having no substantial non-infringing uses. For one, this statement focuses on the dx tool even though the Android SDK is identified as the Accused Instrumentality. Further, setting aside Google's position that all uses of the dx tool are non-infringing, Oracle's contentions are still factually incorrect as the dx tool has a myriad of substantial non-infringing uses, which are not even purported to be infringing by Oracle. The dx tool can translate Dalvik assembly language into Dalvik bytecode; it contains a .class file dumper/disassembler; it can perform partial translations of .class files; it can generate a human-readable dump of.dex files; and it provides a code annotation mode called "annotool." This is merely an exemplary list and an example of how, if Oracle made an effort to meet its burden on this issue, Google could provide a response. Although the bulk of the claim chart refers only to the dx tool, Oracle has changed its allegations to purport to accuse the Android SDK. Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.

- **Claims 1, 6, and 18, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit F the actual performance of any allegedly infringing method and has instead relied on general statements referring to "Android and its development environment . . . ." All of these claims implicate the performance of a method and the charts in Exhibit F are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because it has not identified any allegedly infringing act or purported direct infringer for these claims. Oracle's bare assertion that "[t]he Android code

cited below necessarily infringes because developers must run javac and the Android dx tool to build Android applications and run the Dalvik virtual machine to test them," (*see* Ex. F at 1) does not amount to proof of an actual use that the dx tool and associated code shown in the charts has been executed in the manner alleged.

- **Claims 12, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit F any specific device that allegedly infringes and has instead relied on general statements referring to "[a]ny device or computer which can run the Android dx tool." Oracle has not made a showing of infringement because it has not identified any specific allegedly infringing device or purported direct infringer for these claims and has yet to provide them in supplemental disclosures under the Patent Local Rules. Oracle's bare assertion that "[t]he Android code cited below necessarily infringes because developers must run javac and the Android dx tool to build Android applications and run the Dalvik virtual machine to test them," (*see* Ex. F at 1) does not amount to proof of an actual use that the dx tool and associated code shown in the charts has been executed in a device in the manner alleged.
- **Claim 1 and all dependent claims that depend therefrom:** Oracle accuses its own javac compiler as an element of its allegations for United States Patent No. 6,061,520. Further, to the extent Oracle is relying on a joint infringement theory for these claims, Oracle cannot establish the requisite direction or control. Upon information and belief, Google continues to expect that discovery will reveal that alleged direct infringers, if specifically identified, would be licensed to use that product. Until Oracle identifies on a claim by claim basis the identity of alleged direct infringers performing each step of each claim and Google receives complete production of Oracle's licenses, Google cannot respond completely.
- **All Asserted Claims:** As presently understood, Oracle has not made a showing of infringement of claim 1 at least because the material cited for the "simulating execution of the byte codes of the clinit method against a memory without executing the byte codes to identify the static initialization of the array by the preloader" element at pages 9-19 of Exhibit F does not meet the claim element even if it were implemented and used in a device

in the form it is recited in Exhibit F and Oracle has not made a showing of infringement of claim 6 because the material cited for the "play executing the code without running the code on the component to identify the operation if the code were run by the processing component" element at pages 39-44 of Exhibit F does not meet the claim element even if it were implemented and used in a device in the form it is recited in Exhibit F because it would not employ a method of simulating execution or play executing in that there would be no actual execution of the byte codes while identifying an array initialization instruction. The material cited for this element is a pattern matching algorithm in which bytecodes are not executed and/or in which there is no simulation of execution of the bytecodes. Each other independent claim in Exhibit F references Oracle's citation for claim 1 or claim 6 for a similar element and the same basis applies to those claims. (*See, e.g.*, Claim 12 ("*See* claim 1, *supra*" in chart for "play executing the clinit method . . ." element), Claim 18 ("*See* claim 6, *supra*" in chart for "simulating execution . . ." element).) Although the bulk of the claim chart refers only to the dx tool, Oracle has changed its allegations to purport to accuse the Android SDK. Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.

- **All Asserted Claims:** As presently understood, Oracle has not made a showing of infringement for any of the asserted claims at least because the material cited in Exhibit F for elements requiring creating or storing an instruction to perform a particular function, e.g., Claim 1 ("storing . . . an instruction requesting the static initialization of the array), Claims 6, 18 ("creating an instruction for the processing component to perform the operation"), Claim 12 ("creating an instruction to perform the static initialization"), does not meet the claim elements even if it were implemented and used in a device in the form it is recited in Exhibit F because (1) it would not employ a method that creates or stores a single instruction to perform each of the respective accused functions in that there are multiple instructions identified in Exhibit F, and none of them alone can be used to create and initialize the recited data structure with values contained in the instruction; and/or (2) it would not employ a method that creates or stores a constant pool entry. Although the bulk of the claim chart

refers only to the dx tool, Oracle has changed its allegations to purport to accuse the Android SDK. Google is continuing to perform its own analysis on this change and will supplement these responses as necessary.

- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. The dx tool has a myriad of substantial non-infringing uses, which are not even purported to be infringing by Oracle. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.
- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

The '720 Patent

- **General:** For direct infringement Oracle accuses "device[s] running Android," for system claims, "storage devices containing Android code," for computer-readable medium claims, and "[a]nyone who uses a device running Android code" for method claims. (*See* Ex. G to Oracle's Final Patent L.R. 3-1 Contentions at 2.) Oracle has not provided evidence of actual operation on an actual device, identification of specific storage devices, or evidence of an actual performance of the method. Instead, Oracle states that 1) "Oracle has determined that Android devices execute much of the code cited below every time the devices start up," 2) "Other cited code is invoked when a developer runs the Android Compatibility Test Suite (CTS)," and 3) "Oracle determined that many of these code portions are executed even before a user can interact with a device" based on analysis using the Android Compatibility Test Suite (CTS) and a "mobile device emulator" included with the Android SDK. (*See* Ex. G at 1.) Oracle does not provide any of the factual evidence of this analysis and the analysis does not support the conclusion that Oracle draws regarding the operation of actual devices.

First, even if true, executing "much" of the code, and "many of the[] code portions," cannot demonstrate infringement particularly when Oracle does not provide any break down of what code or code portions it cites in its claim charts are admittedly not executed based in its own analysis. Second, although a trace log of the emulator may show that certain methods are called, it does not mean the CTS checks for the existence of these methods, and, as such, a device could pass compatibility testing with modifications that would result in those methods not being called. Google is continuing to perform its own analysis, which is hampered by Oracle's failure to timely provide any factual basis and continued failure to provide the complete factual basis for its contentions and Google will supplement these responses as necessary.

- **Claims 1 and 20, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit G any specific device that allegedly infringes and has instead relied on general statements referring to "[a] system running Android." Oracle has not made a showing of infringement because it has not identified any specific alleged infringing device or purported direct infringer for these claims, and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.
- **Claim 10, and all dependent claims that depend therefrom:** For these claims, Oracle has failed to identify on a claim by claim basis in Exhibit G the actual performance of any allegedly infringing method and has instead relied on a reference to claim 1, which contains a general statement "[a] system running Android . . . ." Because claim 10 is a method claim, this allegation is deficient on its face. All of these claims implicate the performance of a method and the charts in Exhibit G are devoid of any example of any method being performed, thereby precluding a finding of infringement. Oracle has not made a showing of infringement because it has not identified any alleged infringing act or purported direct infringer for these claims and as discussed above, Oracle's reliance on the CTS and emulator as evidence of actual operation of a device is flawed.

- **All Asserted Claims:** As presently understood, Oracle has not made a showing of infringement of claim 1 even if the materials cited in Exhibit G were implemented and used in a device in the form recited at least because:
    - The identified functionality, e.g., for element 1-c, does not dynamically identify classes to be preloaded.
    - The identified functionality does not both (i) "clone the [master runtime system process] memory space as a child runtime system process" and at the same time also (ii) "defer copying of the memory space of the master runtime system process."
    - The identified fork functionality does not clone the clone the memory space of the master runtime system process as a child runtime system process.

  In view of the fact that Oracle has served its Final Patent L.R. 3-1 Contentions, it is no longer necessary for Google to provide the preceding "if it were implemented" hypothetical response. Because Oracle still has not provided any evidence of actual performance of a method on devices, however, Google is left with only the emulator example to analyze. Google believes the same reasoning applies to the emulator example and will supplement as it continues its investigation.
- **All Asserted Claims:** Oracle has not made any showing or specific allegation of indirect infringement attributable to Google through inducement or contributory infringement. Oracle has not demonstrated that the code identified is actually used to perform a method by any direct infringer, thereby precluding indirect infringement. Further, Oracle has not demonstrated that Google had specific knowledge of this patent sufficient for either inducement or contributory infringement. Oracle's claims of indirect and contributory infringement also fail for reasons cited above under the General Allegations heading.
- **All Asserted Claims:** Google served its Invalidity Contentions on January 18, 2011, detailing its bases for the invalidity of each asserted claim of this patent. Google contends that each asserted claim is invalid and therefore Google cannot infringe such a claim.

All Patents and Asserted Claims

- Oracle has not made any showing or any specific allegation that Google directed and controlled other parties to the extent multiple parties are required to infringe a claim. As presently understood, Oracle cannot make this showing. Because Oracle has served its Final Patent L.R. 3-1 Contentions without identifying specific examples of direct infringement, Google is left with significant ambiguity. For example, it is believed that Oracle's allegations regarding the '520 patent and in particular that developers must run Oracle's javac.exe, implicate joint infringement issues and that Oracle cannot show Google exerts direction and control over Oracle such that Google is a direct infringer as claimed by Oracle.

Google reiterates that the above contentions are being made prematurely in view of Oracle's substantive supplementation three weeks ago—which contained some substantial departures from prior positions held for months—in view of inadequate disclosures by Oracle, as well as in advance of any final claim construction rulings. Google reserves the right to amend and supplement this response as it obtains additional information regarding Oracle's contentions, as well as after any claim construction order.

DATED: April 27, 2011

KING & SPALDING LLP

By: /s/ Scott T. Weingaertner

SCOTT T. WEINGAERTNER (*Pro Hac Vice*)
sweingaertner@kslaw.com
ROBERT F. PERRY
rperry@kslaw.com
BRUCE W. BABER *(Pro Hac Vice)*
bbaber@kslaw.com
1185 Avenue of the Americas
New York, NY 10036-4003
Telephone: (212) 556-2100
Facsimile: (212) 556-2222

DONALD F. ZIMMER, JR. (SBN 112279)
fzimmer@kslaw.com
CHERYL A. SABNIS (SBN 224323)
csabnis@kslaw.com
KING & SPALDING LLP
101 Second Street – Suite 2300
San Francisco, CA 94105
Telephone: (415) 318-1200
Facsimile: (415) 318-1300

IAN C. BALLON (SBN 141819)
ballon@gtlaw.com
HEATHER MEEKER (SBN 172148)
meekerh@gtlaw.com
GREENBERG TRAURIG, LLP
1900 University Avenue
East Palo Alto, CA 94303
Telephone: (650) 328-8500
Facsimile: (650) 328-8508

ATTORNEYS FOR DEFENDANT
GOOGLE INC.

CERTIFICATE OF SERVICE

I hereby certify that on this day, April 27, 2011, I served a true and correct copy of DEFENDANT GOOGLE INC.'S FOURTH SUPPLEMENTAL RESPONSES TO PLAINTIFF'S INTERROGATORIES, SET ONE, NO. 3 via e-mail on the following individuals:

David Boies
Boies Schiller and Flexner
333 Main Street
Armonk, NY 10504
914-749-8201
Fax: 914-749-8300
Email: Dboies@bsfllp.com

Deborah Kay Miller
Oracle USA, Inc Legal Department
500 Oracle Parkway
Redwood Shores, CA 94065
(650) 506-0563
Email: Deborah.Miller@oracle.com

Dorian Estelle Daley
500 Oracle Parkway
Redwood City, CA 94065
(650) 506-5200
Fax: (650) 506-7114
Email: Dorian.daley@oracle.com

Matthew M. Sarboraria
500 Oracle Parkway
Redwood City, CA 94065
Fax: (650) 506-7114
Email: matthew.sarboraria@oracle.com

Steven Christopher Holtzman
Boies, Schiller & Flexner LLP
1999 Harrison Street
Suite 900
Oakland, CA 94612
510-874-1000
Fax: 510-874-1460
Email: Sholtzman@bsfllp.com

Counsel of record for Morrison Foerster:
Michael A. Jacobs
Marc. David Peters
Daniel P. Muino
Roman A. Swoopes
Ruchika Agrawal
Email: OracleMoFoServiceList@mofo.com

April 27, 2011.

/s/ Steven T. Snyder
Steven T. Snyder

Exhibit G

android developers

Download the Android NDK

The Android NDK is a companion tool to the Android SDK that lets you build performance-critical portions of your apps in native code. It provides headers and libraries that allow you to build activities, handle user input, use hardware sensors, access application resources, and more, when programming in C or C++. If you write native code, your applications are still packaged into an .apk file and they still run inside of a virtual machine on the device. The fundamental Android application model does not change.

Using native code does not result in an automatic performance increase, but always increases application complexity. If you have not run into any limitations using the Android framework APIs, you probably do not need the NDK. Read What is the NDK? for more information about what the NDK offers and whether it will be useful to you.

The NDK is designed for use *only* in conjunction with the Android SDK. If you have not already installed and setup the Android SDK, please do so before downloading the NDK.

Platform	Package	Size	MD5 Checksum
Windows	android-ndk-r5c-windows.zip	61627716 bytes	2c7423842fa0f46871eab118495d4b45
Mac OS X (intel)	android-ndk-r5c-darwin-x86.tar.bz2	50714712 bytes	183bfbbd85cf8e4c0bd7531e8803e75d
Linux 32/64-bit (x86)	android-ndk-r5c-linux-x86.tar.bz2	44539890 bytes	7659dfdc97026ed1d913e224d0531f61

Revisions

The sections below provide information and notes about successive releases of the NDK, as denoted by revision number.

▼ Android NDK, Revision 5c *(June 2011)*

This release of the NDK does not include any new features compared to r5b. The r5c release addresses the following problems in the r5b release:

Important bug fixes:

- `ndk-build`: Fixed a rare bug that appeared when trying to perform parallel builds of debuggable projects.
- Fixed a typo that prevented `LOCAL_WHOLE_STATIC_LIBRARIES` to work correctly with the new toolchain and added documentation for this in `docs/ANDROID-MK.html`.
- Fixed a bug where code linked against `gnustl_static` crashed when run on platform releases older than API level 8 (Android 2.2).
- `ndk-gdb`: Fixed a bug that caused a segmentation fault when debugging Android 3.0 or newer devices.
- `<android/input.h>`: Two functions that were introduced in API level 9 (Android 2.3) were incorrect and are fixed. While this breaks the source API, the binary interface to the system is unchanged. The incorrect functions were missing a `history_index` parameter, and the correct definitions are shown

below:

```
float AMotionEvent_getHistoricalRawX(const AInputEvent* motion_event,
                                     size_t pointer_index,
                                     size_t history_index);
float AMotionEvent_getHistoricalRawY(const AInputEvent* motion_event,
                                     size_t pointer_index,
                                     size_t history_index);
```

- Updated the C library ARM binary for API level 9 (Android 2.3) to correctly expose at link time new functions that were added in that API level (for example, `pthread_rwlock_init`).

Minor improvements and fixes:

- Object files are now always linked in the order they appear in `LOCAL_SRC_FILES`. This was not the case previously because the files were grouped by source extensions instead.
- When `import-module` fails, it now prints the list of directories that were searched. This is useful to check that the `NDK_MODULE_PATH` definition used by the build system is correct.
- When `import-module` succeeds, it now prints the directory where the module was found to the log (visible with `NDK_LOG=1`).
- Increased the build speed of debuggable applications when there is a very large number of include directories in the project.
- `ndk-gdb`: Better detection of `adb shell` failures and improved error messages.
- `<pthread.h>`: Fixed the definition of `PTHREAD_RWLOCK_INITIALIZER` for API level 9 (Android 2.3) and higher.
- Fixed an issue where a module could import itself, resulting in an infinite loop in GNU Make.
- Fixed a bug that caused the build to fail if `LOCAL_ARM_NEON` was set to true (typo in `build/core/build-binary.mk`).
- Fixed a bug that prevented the compilation of .s assembly files (`.S` files were okay).

▼ Android NDK, Revision 5b *(January 2011)*

This release of the NDK does not include any new features compared to r5. The r5b release addresses the following problems in the r5 release:

- The r5 binaries required glibc 2.11, but the r5b binaries are generated with a special toolchain that targets glibc 2.7 or higher instead. The Linux toolchain binaries now run on Ubuntu 8.04 or higher.
- Fixes a compiler bug in the arm-linux-androideabi-4.4.3 toolchain. The previous binary generated invalid thumb instruction sequences when dealing with signed chars.
- Adds missing documentation for the "gnustl_static" value for APP_STL, that allows you to link against a static library version of GNU libstdc++.
- The following `ndk-build` issues are fixed:
  - A bug that created inconsistent dependency files when a compilation error occured on Windows. This prevented a proper build after the error was fixed in the source code.
  - A Cygwin-specific bug where using very short paths for the Android NDK installation or the project path led to the generation of invalid dependency files. This made incremental builds impossible.
  - A typo that prevented the cpufeatures library from working correctly with the new NDK toolchain.
  - Builds in Cygwin are faster by avoiding calls to `cygpath -m` from GNU Make for every source or object file, which caused problems with very large source trees. In case this doesn't work properly, define `NDK_USE_CYGPATH=1` in your environment to use `cygpath -m` again.
  - The Cygwin installation now notifies the user of invalid installation paths that contain spaces. Previously, an invalid path would output an error that complained about an incorrect version of GNU

Make, even if the right one was installed.

- Fixed a typo that prevented the `NDK_MODULE_PATH` environment variable from working properly when it contained multiple directories separated with a colon.
- The `prebuilt-common.sh` script contains fixes to check the compiler for 64-bit generated machine code, instead of relying on the host tag, which allows the 32-bit toolchain to rebuild properly on Snow Leopard. The toolchain rebuild scripts now also support using a 32-bit host toolchain.
- A missing declaration for `INET_ADDRSTRLEN` was added to `<netinet/in.h>`.
- Missing declarations for `IN6_IS_ADDR_MC_NODELOCAL` and `IN6_IS_ADDR_MC_GLOBAL` were added to `<netinet/in6.h>`.
- 'asm' was replaced with '__asm__' in `<asm/byteorder.h>` to allow compilation with `-std=c99`.

▼ Android NDK, Revision 5 *(December 2010)*

This release of the NDK includes many new APIs, most of which are introduced to support the development of games and similar applications that make extensive use of native code. Using the APIs, developers have direct native access to events, audio, graphics and window management, assets, and storage. Developers can also implement the Android application lifecycle in native code with help from the new `NativeActivity` class. For detailed information describing the changes in this release, read the `CHANGES.HTML` document included in the downloaded NDK package.

General notes:

- Adds support for native activities, which allows you to implement the Android application lifecycle in native code.
- Adds native support for the following:
  - Input subsystem (such as the keyboard and touch screen)
  - Access to sensor data (accelerometer, compass, gyroscope, etc).
  - Event loop APIs to wait for things such as input and sensor events.
  - Window and surface subsystem
  - Audio APIs based on the OpenSL ES standard that support playback and recording as well as control over platform audio effects
  - Access to assets packaged in an `.apk` file.
- Includes a new toolchain (based on GCC 4.4.3), which generates better code, and can also now be used as a standalone cross-compiler, for people who want to build their stuff with `./configure && make`. See docs/STANDALONE-TOOLCHAIN.html for the details. The binaries for GCC 4.4.0 are still provided, but the 4.2.1 binaries were removed.
- Adds support for prebuilt static and shared libraries (docs/PREBUILTS.html) and module exports and imports to make sharing and reuse of third-party modules much easier (docs/IMPORT-MODULE.html explains why).
- Provides a default C++ STL implementation (based on STLport) as a helper module. It can be used either as a static or shared library (details and usage examples are in sources/android/stlport/README). Prebuilt binaries for STLport (static or shared) and GNU libstdc++ (static only) are also provided if you choose to compile against those libraries instead of the default C++ STL implementation. C++ Exceptions and RTTI are not supported in the default STL implementation. For more information, see docs/CPLUSPLUS-SUPPORT.HTML.
- Includes improvements to the `cpufeatures` helper library that improves reporting of the CPU type (some devices previously reported ARMv7 CPU when the device really was an ARMv6). We recommend developers that use this library to rebuild their applications then upload to Market to benefit from the improvements.
- Adds an EGL library that lets you create and manage OpenGL ES textures and services.

- Adds new sample applications, `native-plasma` and `native-activity`, to demonstrate how to write a native activity.
- Includes many bugfixes and other small improvements; see docs/CHANGES.html for a more detailed list of changes.

▼ Android NDK, Revision 4b *(June 2010)*

NDK r4b notes:

> Includes fixes for several issues in the NDK build and debugging scripts — if you are using NDK r4, we recommend downloading the NDK r4b build. For detailed information describing the changes in this release, read the CHANGES.TXT document included in the downloaded NDK package.

General notes:

- Provides a simplified build system through the new `ndk-build` build command.
- Adds support for easy native debugging of generated machine code on production devices through the new `ndk-gdb` command.
- Adds a new Android-specific ABI for ARM-based CPU architectures, `armeabi-v7a`. The new ABI extends the existing `armeabi` ABI to include these CPU instruction set extensions:
  - Thumb-2 instructions
  - VFP hardware FPU instructions (VFPv3-D16)
  - Optional support for ARM Advanced SIMD (NEON) GCC intrinsics and VFPv3-D32. Supported by devices such as Verizon Droid by Motorola, Google Nexus One, and others.
- Adds a new `cpufeatures` static library (with sources) that lets your app detect the host device's CPU features at runtime. Specifically, applications can check for ARMv7-A support, as well as VFPv3-D32 and NEON support, then provide separate code paths as needed.
- Adds a sample application, `hello-neon`, that illustrates how to use the `cpufeatures` library to check CPU features and then provide an optimized code path using NEON instrinsics, if supported by the CPU.
- Lets you generate machine code for either or both of the instruction sets supported by the NDK. For example, you can build for both ARMv5 and ARMv7-A architectures at the same time and have everything stored to your application's final `.apk`.
- To ensure that your applications are available to users only if their devices are capable of running them, Android Market now filters applications based on the instruction set information included in your application — no action is needed on your part to enable the filtering. Additionally, the Android system itself also checks your application at install time and allows the installation to continue only if the application provides a library that is compiled for the device's CPU architecture.
- Adds support for Android 2.2, including a new stable API for accessing the pixel buffers of `Bitmap` objects from native code.

▼ Android NDK, Revision 3 *(March 2010)*

General notes:

- Adds OpenGL ES 2.0 native library support.
- Adds a sample application,`hello-gl2`, that illustrates the use of OpenGL ES 2.0 vertex and fragment shaders.
- The toolchain binaries have been refreshed for this release with GCC 4.4.0, which should generate slightly more compact and efficient machine code than the previous one (4.2.1). The NDK also still provides the 4.2.1 binaries, which you can optionally use to build your machine code.

▼ Android NDK, Revision 2 *(September 2009)*

Originally released as "Android 1.6 NDK, Release 1".

General notes:

- Adds OpenGL ES 1.1 native library support.
- Adds a sample application, `san-angeles`, that renders 3D graphics through the native OpenGL ES APIs, while managing activity lifecycle with a `GLSurfaceView` object.

▼ Android NDK, Revision 1 *(June 2009)*

Originally released as "Android 1.5 NDK, Release 1".

General notes:

- Includes compiler support (GCC) for ARMv5TE instructions, including Thumb-1 instructions.
- Includes system headers for stable native APIs, documentation, and sample applications.

Installing the NDK

Installing the NDK on your development computer is straightforward and involves extracting the NDK from its download package.

Before you get started make sure that you have downloaded the latest Android SDK and upgraded your applications and environment as needed. The NDK is compatible with older platform versions but not older versions of the SDK tools. Also, take a moment to review the System and Software Requirements for the NDK, if you haven't already.

To install the NDK, follow these steps:

1. From the table at the top of this page, select the NDK package that is appropriate for your development computer and download the package.
2. Uncompress the NDK download package using tools available on your computer. When uncompressed, the NDK files are contained in a directory called `android-ndk-<version>`. You can rename the NDK directory if necessary and you can move it to any location on your computer. This documentation refers to the NDK directory as `<ndk>`.

You are now ready to start working with the NDK.

Getting Started with the NDK

Once you've installed the NDK successfully, take a few minutes to read the documentation included in the NDK. You can find the documentation in the `<ndk>/docs/` directory. In particular, please read the OVERVIEW.HTML document completely, so that you understand the intent of the NDK and how to use it.

If you used a previous version of the NDK, take a moment to review the list of NDK changes in the CHANGES.HTML document.

Here's the general outline of how you work with the NDK tools:

1. Place your native sources under `<project>/jni/...`
2. Create `<project>/jni/Android.mk` to describe your native sources to the NDK build system
3. Optional: Create `<project>/jni/Application.mk`.
4. Build your native code by running the 'ndk-build' script from your project's directory. It is located in the top-level NDK

directory:

```
cd <project>
<ndk>/ndk-build
```

The build tools copy the stripped, shared libraries needed by your application to the proper location in the application's project directory.

5. Finally, compile your application using the SDK tools in the usual way. The SDK build tools will package the shared libraries in the application's deployable `.apk` file.

For complete information on all of the steps listed above, please see the documentation included with the NDK package.

Sample Applications

The NDK includes sample Android applications that illustrate how to use native code in your Android applications. For more information, see Sample Applications.

Discussion Forum and Mailing List

If you have questions about the NDK or would like to read or contribute to discussions about it, please visit the android-ndk group and mailing list.

Exhibit H

FILED UNDER SEAL

Exhibit I

FILED UNDER SEAL

Exhibit J

FILED UNDER SEAL

Exhibit K

FORM 10-K

ORACLE CORP - ORCL

Filed: July 01, 2010 (period: May 31, 2010)

Annual report which provides a comprehensive overview of the company for the past year

UNITED STATES
SECURITIES AND EXCHANGE COMMISSION
Washington, D.C. 20549

FORM 10-K

☒ **ANNUAL REPORT PURSUANT TO SECTION 13 OR 15(d) OF THE SECURITIES EXCHANGE ACT OF 1934**

For the fiscal year ended May 31, 2010

OR

☐ **TRANSITION REPORT PURSUANT TO SECTION 13 OR 15(d) OF THE SECURITIES EXCHANGE ACT OF 1934**

For the transition period from _____ to _____

Commission file number: 000-51788

Oracle Corporation

(Exact name of registrant as specified in its charter)

Delaware	**54-2185193**
(State or other jurisdiction of incorporation or organization)	**(I.R.S. Employer Identification No.)**
500 Oracle Parkway Redwood City, California	**94065**
(Address of principal executive offices)	**(Zip Code)**

(650) 506-7000
(Registrant's telephone number, including area code)

Securities registered pursuant to Section 12(b) of the Act:

Title of each class	**Name of each exchange on which registered**
Common Stock, par value $0.01 per share	The NASDAQ Stock Market LLC

Securities registered pursuant to Section 12(g) of the Act:
None

Indicate by check mark if the registrant is a well-known seasoned issuer, as defined in Rule 405 of the Securities Act. YES ☒ NO ☐

Indicate by check mark if the registrant is not required to file reports pursuant to Section 13 or Section 15(d) of the Act. YES ☐ NO ☒

Indicate by check mark whether the registrant (1) has filed all reports required to be filed by Section 13 or 15(d) of the Securities Exchange Act of 1934 during the preceding 12 months (or for such shorter period that the registrant was required to file such reports), and (2) has been subject to such filing requirements for the past 90 days. YES ☒ NO ☐

Indicate by check mark whether the registrant has submitted electronically and posted on its corporate Web site, if any, every Interactive Data File required to be submitted and posted pursuant to Rule 405 of Regulation S-T (§232.405 of this chapter) during the preceding 12 months (or for such shorter period that the registrant was required to submit and post such files). YES ☒ NO ☐

Indicate by check mark if disclosure of delinquent filers pursuant to Item 405 of Regulation S-K (§229.405 of this chapter) is not contained herein, and will not be contained, to the best of registrant's knowledge, in definitive proxy or information statements incorporated by reference in Part III of this Form 10-K or any amendment to this Form 10-K. ☐

Indicate by check mark whether the registrant is a large accelerated filer, an accelerated filer, a non-accelerated filer, or a smaller reporting company. See the definitions of "large accelerated filer," "accelerated filer" and "smaller reporting company" in Rule 12b-2 of the Exchange Act.

Large accelerated filer ☒	Accelerated filer ☐
Non-accelerated filer ☐ (Do not check if a smaller reporting company)	Smaller reporting company ☐

Indicate by check mark whether the registrant is a shell company (as defined in Rule 12b-2 of the Exchange Act). YES ☐ NO ☒

The aggregate market value of the voting stock held by non-affiliates of the registrant was $85,417,126,000 based on the number of shares held by non-affiliates of the registrant as of May 31, 2010, and based on the closing sale price of common stock as reported by the NASDAQ Global Select Market on November 30, 2009, which is the last business day of the registrant's most recently completed second fiscal quarter. This calculation does not reflect a determination that persons are affiliates for any other purposes.

Number of shares of common stock outstanding as of June 22, 2010: 5,026,247,000.

Documents Incorporated by Reference:

Portions of the registrant's definitive proxy statement relating to its 2010 annual stockholders' meeting are incorporated by reference into Part III of this Annual Report on Form 10-K where indicated.

ORACLE CORPORATION

FISCAL YEAR 2010
FORM 10-K
ANNUAL REPORT

TABLE OF CONTENTS

Cautionary Note on Forward-Looking Statements

For purposes of this Annual Report, the terms "Oracle," "we," "us" and "our" refer to Oracle Corporation and its consolidated subsidiaries. This Annual Report on Form 10-K contains statements that are not historical in nature, are predictive in nature, or that depend upon or refer to future events or conditions or contain forward-looking statements within the meaning of Section 21 of the Securities Exchange Act of 1934, as amended, and the Private Securities Litigation Reform Act of 1995. Forward-looking statements may be preceded by, followed by or include the words "expects," "anticipates," "intends," "plans," "believes," "seeks," "estimates," "will," "is designed to" and similar expressions. We claim the protection of the safe harbor for forward-looking statements contained in the Private Securities Litigation Reform Act of 1995 for all forward-looking statements. We have based these forward-looking statements on our current expectations and projections about future events. These forward-looking statements are subject to risks, uncertainties, and assumptions about our business. Factors that might cause or contribute to such differences include, but are not limited to, those discussed in "Risk Factors" beginning on page 17 of this Annual Report, and as may be updated in filings we make from time to time with the Securities and Exchange Commission (the "SEC"). You should understand that the following important factors, in addition to those discussed elsewhere in this Annual Report, could affect our future results, and could cause those results or other outcomes to differ materially from those expressed or implied in the forward-looking statements:

- Economic, political and market conditions, including the recent recession and global economic crisis, can adversely affect our business, results of operations and financial condition, including our revenue growth and profitability, which in turn could adversely affect our stock price.

- We may fail to achieve our financial forecasts due to such factors as delays or size reductions in transactions, fewer large transactions in a particular quarter, unanticipated fluctuations in currency exchange rates, delays in delivery of new products or releases or a decline in our renewal rates for software license updates and product support.

- Our entrance into the hardware systems business may not be successful, and we may fail to achieve our financial forecasts with respect to this new business.

- We have an active acquisition program and our acquisitions, including our acquisition of Sun Microsystems, Inc. may not be successful, may involve unanticipated costs or other integration issues or may disrupt our existing operations.

- Our international sales and operations subject us to additional risks that can adversely affect our operating results, including risks relating to foreign currency gains and losses and risks relating to compliance with international and U.S. laws that apply to our international operations.

- Intense competitive forces demand rapid technological advances and frequent new product introductions and could require us to reduce prices or cause us to lose customers.

- If we are unable to develop new or sufficiently differentiated products and services, or to enhance and improve our products and support services in a timely manner or to position and/or price our products and services to meet market demand, customers may not buy new software licenses or hardware systems products or purchase or renew support contracts.

We have no obligation to publicly update or revise any forward-looking statements, whether as a result of new information, future events or risks, except to the extent required by applicable securities laws. If we do update one or more forward-looking statements, no inference should be drawn that we will make additional updates with respect to those or other forward-looking statements. New information, future events or risks could cause the forward-looking events we discuss in this Annual Report not to occur. You should not place undue reliance on these forward-looking statements, which reflect our opinions only as of the date of this Annual Report. You should carefully review the risk factors described in other documents we file from time to time with the SEC, including the Quarterly Reports on Form 10-Q to be filed by us in our 2011 fiscal year, which runs from June 1, 2010 to May 31, 2011.

PART I

Item 1. Business

General

We are the world's largest enterprise software company. As a result of our acquisition of Sun Microsystems, Inc. (Sun) in January 2010, we are also a leading provider of hardware products and services. We develop, manufacture, market, distribute and service database and middleware software, applications software and hardware systems, consisting primarily of computer server and storage products, which are designed to help our customers manage and grow their business operations.

Our goal is to be the world's most complete, open and integrated enterprise software and hardware company. We offer customers scalable, reliable, secure and integrated software and hardware solutions that are designed to improve transactional efficiencies, adapt to an organization's unique needs and allow better ways to access and manage information and automate business processes at a lower total cost of ownership. We seek to be an industry leader in each of the specific product categories in which we compete and to expand into new and emerging markets.

We believe our internal, or organic, growth and continued innovation with respect to our software, hardware and services businesses are the foundation of our long-term strategic plan. In fiscal 2010, 2009 and 2008 we invested $3.3 billion, $2.8 billion and $2.7 billion, respectively, in research and development to enhance our existing portfolio of products and services and to develop new products, features and services. We also believe that an active acquisition program is an important element of our corporate strategy as it strengthens our competitive position, expands our customer base, provides greater scale to accelerate innovation, grows our revenues and earnings, and increases stockholder value. In recent years, we have invested billions of dollars to acquire a number of companies, products, services and technologies that add to, are complementary to or have otherwise enhanced our existing offerings. We expect to continue to acquire companies, products, services and technologies in furtherance of our corporate strategy.

Oracle Corporation was incorporated in 2005 as a Delaware corporation and is the successor to operations originally begun in June 1977.

Software, Hardware Systems and Services

With the acquisition of Sun, we are now organized into three businesses—software, hardware systems and services—which are further divided into seven operating segments. Our software business is comprised of two operating segments: (1) new software licenses and (2) software license updates and product support. Our hardware systems business consists of two operating segments: (1) hardware systems products and (2) hardware systems support. Our services business is comprised of three operating segments: (1) consulting, (2) On Demand and (3) education.

Our software, hardware systems and services businesses represented 77%, 9% and 14% of our total revenues, respectively, in fiscal 2010. Prior to our acquisition of Sun in fiscal 2010, we did not have a hardware systems segment. Our software and services businesses represented 81% and 19% of our total revenues, respectively, in fiscal 2009 and 80% and 20% of our total revenues, respectively, in fiscal 2008. See Note 16 of Notes to Consolidated Financial Statements for additional information related to our operating segments.

Software Business

Our software business consists of our new software licenses segment and software license updates and product support segment.

New Software Licenses

The new software licenses operating segment of our software business includes the licensing of database and middleware software as well as applications software. As a result of our acquisition of Sun, we acquired software technologies that expanded and enhanced our existing database and middleware software product offerings, including the Java technology platform.

Our software solutions are designed to help customers reduce the cost and complexity of their information technology (or IT) infrastructures by delivering solutions via a standards-based, or "open", integrated architecture, which allows our software products to work in customer environments that may include Oracle or non-Oracle hardware or software components. This approach supports customer choice, reduces customer risk and can be adapted to the specific needs of any industry or application. In this model, our database and certain of our middleware offerings are designed to manage and protect a customer's underlying business information, while application servers run enterprise applications that are designed to automate multiple business functions and provide intelligence in critical functional areas. Our software products are designed to operate on both single server and clustered server configurations, which we refer to as "grid" software, to support a choice of operating systems, including Solaris, Linux, Microsoft Windows and UNIX.

New software license revenues include fees earned from granting customers licenses to use our software products and exclude revenues derived from software license updates and product support. The standard end user software license agreement for our products generally provides for an initial fee to use the software product in perpetuity based on a maximum number of processors, named users or other metrics. We also have other types of software license agreements, including licenses that are restricted by the number of employees, licenses that provide for a limited term and open source licenses. New software license revenues represented 28%, 31%, and 34% of total revenues in fiscal 2010, 2009 and 2008, respectively.

Database and Middleware Software

Our database and middleware software offerings are designed to provide a cost-effective, high-performance platform for running and managing business applications for small and mid-size businesses, as well as large, global enterprises. Our customers are increasingly focused on reducing the total cost of their IT infrastructure and we believe that our software offerings help them achieve this goal. Our software is designed to accommodate demanding, non-stop business environments, using server, storage and application grids. These grids are designed to scale incrementally as required to address our customers' IT capacity; satisfy their planning and procurement needs; support all of their business applications; lower their overall hardware investment; reduce their risk of data loss and IT infrastructure downtime; and efficiently utilize available IT resources to meet quality of service expectations. New software license revenues from database and middleware products represented 72% of our new software license revenues in both fiscal 2010 and 2009 and 68% of our new software license revenues in fiscal 2008.

Database

The Oracle Database is the world's most popular enterprise database and is designed to enable the secure storage, retrieval and manipulation of all forms of data, including transactional data, business application data, analytic data, and unstructured data in the form of XML files, office documents, images, video, spatial and other specialized forms of data such as human genomic and medical data. The Oracle Database is used for a variety of purposes including online transaction processing, data warehousing, as a document repository or specialized data store and as a database with packaged applications.

The Oracle Database is available in four editions: Express Edition, Standard Edition One, Standard Edition and Enterprise Edition. All editions are built using the same underlying code, which means that our database software can scale from small, single processor servers to clusters of multi-processor servers. A number of optional offerings are available with Oracle Database Enterprise Edition to address specific customer requirements in the areas of performance and scalability, high availability, data security and compliance, data warehousing, information management and systems management. Examples of these options include:

- Oracle Real Application Clusters, which is designed to enable any Oracle Database application to share more efficiently the processing power and memory capacity of a fault tolerant cluster of servers;

- Oracle Exadata, which is designed to improve database storage and query performance by, among other things, enabling query processing closer to customer data and reducing required disk space;

- Oracle Advanced Compression, which is designed to enable customers to reduce the amount of disk space required to store all their business information and improve query performance;

- Oracle Partitioning, which is designed to break down large database tables into smaller segments for faster query performance and easier management of data throughout its lifecycle;

- Oracle Database Security solutions, which are designed to protect data while it is moving over the network to the database (Advanced Security Option), to protect data when it is within the database (Label Security), and to protect data when it is archived to tape (Secure Backup);

- Oracle Database Vault, which is designed to pro-actively safeguard application data stored in the Oracle Database from being accessed by system administrators and other privileged database users to meet regulatory mandates and improve data security;

- Oracle Audit Vault, which is designed to reduce the cost and complexity of compliance reporting and detection of unauthorized activities by automating the collection, consolidation and analysis of enterprise audit data;

- Oracle Active Data Guard, which is designed to improve database performance and reliability by offloading resource-intensive activities from a production database to one or more synchronized standby databases;

- Oracle Spatial and Oracle Locator, which are designed to manage geospatial data and provide the facilities to location enable business applications with advanced geographic information system (GIS) capabilities; and

- Oracle In-Memory Database Cache, which is designed to improve application performance by caching or storing critical parts of Oracle Database in the main memory of the application tier.

In addition to the four editions of the Oracle Database, we also offer a portfolio of specialized database products to address particular customer requirements:

- MySQL, which we acquired as part of our acquisition of Sun, is one of the world's most popular open source databases that is designed to offer developers a simple to use, horizontally-scalable database often used to power high-volume applications such as websites and web-based applications;

- Oracle TimesTen In-Memory Database, which is a memory-optimized relational database that is designed to deliver low latency and high throughput for applications requiring real-time performance in industries such as communications, financial services and defense; and

- Oracle Berkeley DB, which is a family of open source, embeddable, non-relational databases that is designed to allow developers to incorporate a fast, scalable and reliable database engine within their applications and devices.

Middleware

Oracle Fusion Middleware is a broad family of application infrastructure products that is designed to form a reliable and scalable foundation on which customers can build, deploy, secure, access and integrate business applications and automate their business processes. Built on the Java technology platform, Oracle Fusion Middleware suites and products can be used as a foundation for custom, packaged and composite applications.

Oracle Fusion Middleware is designed to protect customers' IT investments and work with both Oracle and non-Oracle database, middleware and applications products through its "hot-pluggable" architecture (which enables customers to easily install and use Oracle Fusion Middleware products within their existing IT environments) and adherence to industry standards such as Java Enterprise Edition (Java EE, formerly J2EE) and Business Process Execution Language (BPEL), among others. By using Oracle Fusion Middleware, we believe our customers can better adapt to business changes rapidly, reduce their risks related to security and compliance, increase user productivity and drive better business decisions. Specifically, Oracle Fusion Middleware is designed to enable customers to integrate heterogeneous business applications, automate business processes, scale applications to meet customer demand, simplify security and compliance, manage lifecycles of documents and get actionable, targeted business intelligence, while continuing to utilize their existing IT systems. In addition, Oracle Fusion Middleware supports multiple development languages and tools, which allows

developers to build and deploy web services, web sites, portals and web-based applications. Oracle Fusion Middleware is used to support Oracle applications, other enterprise applications, independent software vendors that build their own applications and business processes that span multiple application environments.

Oracle Fusion Middleware is available in various products and suites, including the following functional areas:

- Application Server and Application Grid;
- Service-Oriented Architecture and Business Process Management;
- Data Migration and Integration;
- Business Intelligence;
- Identity and Access Management;
- Content Management;
- Portals and User Interaction; and
- Development Tools.

Application Server and Application Grid

The foundation of Oracle Fusion Middleware is Oracle WebLogic Server—an application server that is compliant with the Java EE specification. Designed for grid computing, Oracle WebLogic Server incorporates clustering and caching technology, which increases application reliability, performance, security and scalability. Oracle JRockit is a high performance Java Virtual Machine designed to run Java applications on multi-core processors with higher and more predictable performance. Oracle Coherence is an in-memory data grid solution designed to reduce latency, and improve performance and scalability of business applications by allowing applications to access data in-memory. Oracle Tuxedo runs legacy, mainframe, and non-Java applications written in the C, C++, and COBOL languages with transaction reliability, scale, and performance requirements. The addition of Oracle GlassFish Server from the acquisition of Sun enhances the value of Oracle Fusion Middleware for developers by accelerating development practices and decreasing application time-to-market.

Service-Oriented Architecture Suite and Business Process Management

Service-Oriented Architecture (SOA) is a software development and architecture methodology that creates a modular, re-usable approach to applications development; makes it easier to integrate systems with each other; and reduces the need for costly custom development. Oracle SOA Suite is a suite of middleware software products used to create, deploy, and manage applications on a Service-Oriented Architecture including Oracle JDeveloper, Oracle BPEL Process Manager, Oracle Web Services Manager, Oracle Business Rules, Oracle Business Activity Monitoring, and Oracle Service Bus. Oracle Business Process Management Suite is a suite of software designed to enable business and IT professionals to design, implement, automate, and evolve business processes and workflow within and across organizations. Oracle SOA Governance is designed to maintain the security and integrity of our customers' SOA deployments.

Data Migration and Integration

Oracle's Data Integration offerings consist of Oracle GoldenGate, Oracle Data Integrator, and Oracle Data Quality solutions. Oracle GoldenGate is a high performance data movement and continuous availability solution designed to capture transaction records on one system and to move and apply them to other systems with very low impact on system and network performance. Oracle Data Integrator is an extract-transform-load (ETL) solution that enables users to extract data from one system, transform it from the source system's format to a target system's format, and load it into the target system (such as a data warehouse). Oracle Data Quality enables users to profile data and to clean it using a variety of automated matching and cleansing rules making the data more reliable and more accurate.

Business Intelligence

Oracle Business Intelligence is a comprehensive set of analytic products designed to provide customers with the information they need to make better business decisions. Oracle's Business Intelligence technology consists of Oracle BI Suite Enterprise Edition, a comprehensive query and analysis server; Oracle Essbase, an online analytical processing (OLAP) server; Oracle BI Publisher, a self-service production and operational reporting tool; and Oracle Real-Time Decisions, a real-time data classification and optimization solution. Users can access these tools from a variety of user interfaces including browser-based interactive dashboards; ad hoc query and analysis; proactive detection and alerts integrated with e-mail; Microsoft Office integration including support for Excel, Word, and PowerPoint; and mobile analytics for mobile and smart phones.

Identity and Access Management

Oracle's Identity and Access Management products are designed to enable customers to manage internal and external users, secure corporate information from potential threats and streamline compliance initiatives while lowering the total cost of their security and compliance initiatives. Our offerings include a lightweight directory access protocol (LDAP) directory service to store and manage user identities and policies; identity provisioning to provision users and roles in multiple enterprise applications and systems; access management to manage access control and entitlements for customers, partners, and employees; and identity analytics products to audit and identify users attempting to access systems for which they are not authorized.

Content Management

Oracle's Content Management Suite is an enterprise content management platform allowing users to capture, manage, and publish information that is either unstructured, not easily readable or has not been stored, including documents, images, audio, video, and a wide variety of other forms of digital content. Our Content Management Suite provides customers with a highly scalable document management repository; web content management to publish information to websites and portals; digital asset management to manage and deliver digital content; imaging and process management to capture and process paper documents and document related business processes; and records management to archive and retain documents and electronic records. Oracle Content Management is integrated with business applications to automatically capture and process electronic and paper documents such as invoices, accounts receivable receipts, and sales order documents from these applications.

Portals and User Interaction

Oracle WebCenter Suite is a standards-based enterprise portal product that enables external and employee users to efficiently find the information they need from websites and business applications within the organization; to create collaborative websites, online workspaces, discussion forums, integrated real-time presence and web conferencing to share information with others; and to use a variety of emerging technologies such as really simple syndication (RSS) feeds, tag clouds, linking and search to personalize information delivery. Oracle WebCenter Suite can be used to build a variety of web-based systems including extranet websites, intranet portals, task-oriented collaborative applications, and team communities.

Developer Tools

Oracle JDeveloper is an integrated software environment that is designed to facilitate rapid development of a variety of different types of applications using Oracle Fusion Middleware and popular open source technologies. Oracle JDeveloper provides support for developing Java applications; web services, composite SOA applications and business processes; rich user interfaces using AJAX/DHTML and Flash technologies; and websites using popular scripting languages. Oracle JDeveloper also provides comprehensive application lifecycle management facilities including modeling, building, debugging, unit testing, profiling, and optimizing applications and is integrated with the Oracle Application Development Framework, which provides a declarative framework for building business applications, and popular open source tools including Eclipse and NetBeans.

We also offer a selection of products that are complementary to our database and middleware products, including Oracle Enterprise Manager.

Oracle Enterprise Manager

Oracle Enterprise Manager is Oracle's unified enterprise IT management solution designed to provide integrated systems management; to manage the entire Oracle software stack from applications to disk, and to deliver business-driven applications management. Oracle Enterprise Manager manages all Oracle software products including Oracle Enterprise Linux, Oracle VM, Oracle Database, Fusion Middleware, and business applications. Enterprise Manager also manages Oracle software across its entire lifecycle, including installing and configuring the software; monitoring service levels and performance; diagnosing and troubleshooting problems; configuring and patching software; and providing change management in a unified way across groups of computers or grids and across physical and virtualized environments. We are also integrating Sun's OpsCenter, a product designed by Sun to manage the Solaris operating system, firmware, and storage, with Oracle Enterprise Manager to provide a single management solution to manage hardware and software.

Oracle Fusion Middleware products and applications are built on our Java technology platform.

Java

Java is the computer industry's most widely-used software development language and is viewed as a global standard. For customers, the Java platform is designed to enable a variety of compatible applications, independent of their vendor, and a global community of Java developers, support engineers, and knowledge bases which can help customers reduce the risk and time to deployment as well as the ongoing cost of ownership and maintenance. Java technology is designed to allow developers and vendors to write software on one platform and run it on many different platforms, independent of operating system and hardware architecture. Java technology is licensed to run applications for use in a broad range of systems – embedded devices; SIM cards; mobile phones and smart-phones; IP and Blu-Ray TVs; desktop computers; and server class computers. We expect to continue to innovate and invest in Java technology for the benefit of customers and the Java community.

Applications Software

Our applications software offerings are designed using an open, integrated architecture to manage and automate customers' core business functions, support customer choice, help reduce risk, cost and complexity of customers' IT infrastructures, and enable customers to differentiate their businesses using our technologies. Through a focused strategy of investments in organic research and development and strategic acquisitions, we also provide industry-specific solutions for customers in over 20 industries, including communications, engineering and construction, financial services, health services, manufacturing, public sector, retail and utilities. New software license revenues from applications software represented 28% of our new software license revenues in both fiscal 2010 and 2009 and 32% of new software license revenues in fiscal 2008.

Central to our applications strategy is our Applications Unlimited program, which is our commitment to offer customers that purchase software license updates and product support contracts a choice as to when they wish to upgrade to the next generation of the products they own. Until our customers reach a decision to upgrade to the next generation of the products they own, we protect their investments in their applications by offering them the ability to purchase software license updates and product support contracts for their existing products. Our applications are designed to help customers extend the benefits of their IT investments in our applications, to reduce their investment risk, and to support their evolution to the next generation of enterprise software that best fits their needs. For example, our Oracle Application Integration Architecture provides an open framework for creating adaptable, cross-application business processes. In addition, our applications software products are offered as integrated suites or available on a component basis, and all are built on open architectures that are designed for flexible configuration and open, multi-vendor integration. Our applications are also available in multiple languages and support a broad range of location specific requirements, enabling companies to support both global and local business practices and legal requirements.

We strive to protect our customers' investments in Oracle applications by delivering new product releases that incorporate customer-specific and industry-specific innovations across product lines. Since announcing our Applications Unlimited program, we have delivered major releases on all applications product lines. Our

applications software products combine business functionality with innovative technologies such as role-based analytics, secure search, identity management, self-service and workflow to deliver adaptive industry processes, business intelligence and insights, and optimal end-user productivity. Our applications software products enable efficient management of core business functions, including:

- Enterprise Resource Planning (ERP);
- Customer Relationship Management (CRM);
- Enterprise Performance Management (EPM);
- Supply Chain Management (SCM);
- Business Intelligence Applications (Analytic Applications);
- Enterprise Project Portfolio Management (EPPM); and
- Industry-Specific Applications.

Enterprise Resource Planning (ERP)

Companies use our ERP applications to automate and integrate a variety of their key global business processes, including: supply chain planning, manufacturing, logistics, order fulfillment, asset lifecycle management, purchasing, accounts receivable and payable, general ledger, cash and treasury management, travel and expense management, human resources, payroll, benefits, and talent management. Our ERP applications combine business functionality with innovative technologies such as self-service applications to enable companies to lower the cost of their business operations by providing their customers, suppliers and employees with self-service access to both transaction processing and critical business information.

Customer Relationship Management (CRM)

We offer a suite of CRM applications that help our customers manage their selling processes more efficiently, integrate marketing campaigns and content into their selling processes more effectively, and deliver high quality customer service across multiple access channels, including call centers, websites and mobile devices. Our CRM products also provide many industry-specific features designed to support the specialized needs of users in key sectors, such as communications, high technology, life sciences, financial services, insurance, consumer products, and the public sector.

Enterprise Performance Management (EPM)

We offer a suite of EPM applications to automate, integrate, and administer a broad range of financial and operational management processes within an organization. Our EPM applications enable organizations to define and model their financial structure; to define their operating plans and manage financial budgets; to allocate indirect revenues and costs to better understand business unit profitability; to consolidate and aggregate financial results from a variety of systems; and to manage the financial close and statutory reporting processes. Our EPM applications are integrated with ERP applications; enable improved business decision making; and align operational processes with an organization's financial strategy.

Supply Chain Management (SCM)

We offer a comprehensive set of SCM solutions that span from demand management, sales and operations planning and supply chain planning to product development, manufacturing, transportation and warehouse management. Customers can use Oracle SCM products to predict demand and market requirements, manage the lifecycle of their products, innovate in response to volatile market conditions, align operations across global networks and deploy lean, mixed-mode manufacturing with integrated manufacturing execution systems that meet both discrete and process requirements.

Business Intelligence Applications (Analytic Applications)

We also provide packaged Business Intelligence Applications for ERP, SCM, and CRM processes and industry-specific analytic applications with packaged data models; packaged ETL processes; packaged key performance

indicators (KPIs); and packaged dashboards and scorecards to deliver insight that is tailored for business processes in 20 industries. Our Business Intelligence Applications are built on Oracle's Business Intelligence technology and source data from multiple versions of Oracle CRM, ERP, EPM, and SCM applications as well as from non-Oracle ERP applications. Our EPM and Business Intelligence Applications together with our Business Intelligence technology allow us to offer our customers an integrated solution spanning planning and budgeting; financial management; operational analytics; and reporting.

Enterprise Project Portfolio Management (EPPM)

Our EPPM software solutions target project-intensive industries such as engineering and construction, aerospace and defense, utilities, oil and gas, manufacturing, professional services and project-intensive departments within other industries. Our EPPM solutions help companies propose, prioritize and select project investments and plan, manage and control the most complex projects and project portfolios.

Industry-Specific Applications

Our applications can be tailored to offer customers a variety of industry-specific solutions. As a part of our strategy, we strive to ensure that our applications portfolio addresses the major industry-influenced technology challenges of customers in key industries that we view as strategic to our future growth, including banking and financial services, communications, education, engineering and construction, health sciences, insurance, manufacturing, professional services, public sector, retail and utilities.

Software License Updates and Product Support

We seek to protect and enhance our customers' current investments in Oracle software by offering proactive and personalized support services, including our Lifetime Support policy, and unspecified product enhancements and upgrades. Software license updates provide customers with rights to unspecified software product upgrades and maintenance releases and patches released during the term of the support period. Product support includes internet and telephone access to technical support personnel located in our global support centers, as well as internet access to technical content through "My Oracle Support". Software license updates and product support contracts are generally priced as a percentage of the net new software license fees. Substantially all of our customers purchase software license updates and product support contracts when they acquire new software licenses and renew their software license updates and product support contracts annually. Our software license updates and product support revenues represented 49%, 50% and 46% of our total revenues in fiscal 2010, 2009 and 2008, respectively.

Hardware Systems Business

As a result of our acquisition of Sun in January 2010, we entered into a new hardware systems business. Our hardware systems business consists of two operating segments: hardware systems products and hardware systems support.

Hardware Systems Products

Our customers demand a broad set of hardware systems solutions to manage growing amounts of data and computational requirements, to meet increasing compliance and regulatory demands, and to reduce energy, space, and operational costs. To meet these demands, we have a wide variety of innovative hardware systems offerings, including servers and storage products, networking components, operating systems and other hardware-related software. Our hardware systems products are designed to be "open," or to work in customer environments that may include other Oracle or non-Oracle hardware or software components. We have also designed our hardware systems products to create performance and operational cost advantages for customers when they use our hardware and software together. By combining our server and storage hardware with our software, our open, integrated products better address customer requirements for performance, scalability, reliability, security, ease of management and lower total cost of ownership. Our hardware systems products represented 6% of our total revenues in fiscal 2010.

Servers

We offer a wide range of server systems using our SPARC microprocessor. Our SPARC servers are differentiated by their size, cost, form factor or configuration (rack, blade or stand-alone systems) and customer environments that they target (general purpose or specialized systems). Our mid- and large-size servers are designed to offer greater performance and lower total cost of ownership than mainframe systems for business critical applications and for customers having more computationally intensive needs. Our SPARC servers run the Solaris operating system and are designed for the most demanding mission critical enterprise environments at any scale. We have a long-standing relationship with Fujitsu Limited for the development, manufacturing and marketing of certain of our SPARC server components and products.

We also offer a wide range of x86 servers differentiated by the same features as our SPARC servers. These x86 systems are primarily based on microprocessor platforms from Intel Corporation (Intel) and are also compatible with Solaris, Linux, Windows and other operating systems.

We offer a line of products aimed at the unique needs of original equipment manufacturers (OEMs) and network equipment providers (NEPs). Rack-optimized systems and our blade product offerings combine high-density hardware architecture and system management software that OEMs find particularly useful in building their own solution architectures. Our NEP-certified Sun Netra systems are designed to meet the specialized needs of NEPs.

Storage

Our broad range of storage products are designed to securely manage, protect, archive and restore customers' mission critical data assets and consist of tape, disk, hardware-related software and networking for mainframe and open systems environments. Our storage solutions are designed to improve data availability by providing fast data access and dynamic data protection for restoration and secure archiving for compliance.

Our tape storage product line includes StorageTEK libraries, drives, virtualization systems, media and device software. These products are intended to provide robust solutions for both long-term preservation and near-term protection, of customer data at a lower total cost of ownership.

Our disk storage product lines include data center arrays, mid-range arrays, unified storage, network attached storage, and entry level systems. We also offer software for management and efficient resource utilization and virtualization of storage resources.

Solaris Operating System and Other Hardware-Related Software

Our Solaris operating system is designed to provide a more reliable, secure and scalable operating system environment than other enterprise operating systems through significant core feature development in kernel, networking, security, and file system technologies as well as close integration with hardware features. This provides us with an ability to combine Solaris with our own hardware components to achieve certain performance and efficiency advantages in comparison to our competitors. The Solaris operating system is based on the Unix operating system, but is unique among Unix systems in that it is available on our SPARC servers and x86 servers that include microprocessors from either Intel or Advanced Micro Devices, Inc. We also support Solaris deployed on other companies' hardware products. In addition to Solaris, we also develop a range of hardware-related software, including development tools, compilers, management tools for servers and storage, diagnostic tools, virtualization, and file systems.

Oracle Exadata

Oracle Exadata is designed to be a family of integrated software and hardware products that combine our storage software with our server, storage, and networking components to provide our customers with improved performance at a lower total cost of ownership. For example, our current Oracle Database Machine offers improved performance, scalability and reliability through an integrated architecture featuring increased data bandwidth and localized query processing. Additional options are also available, including Oracle Exadata Hybrid Columnar Compression, which is designed to reduce the disk space required for primary, standby and backup databases.

Networking

We create networking components and products designed to efficiently connect and deploy server and storage clusters in data centers. The development of our networking products includes both hardware and software development for the Infiniband and Ethernet technologies that are used with our server and storage products and are integrated into our management tools.

Hardware Systems Support

Customers that purchase our hardware systems products may also elect to purchase our hardware systems support offerings. Our hardware systems support offerings provide customers with software updates for the software components that are essential to the functionality of our systems and storage products, such as Solaris, and can include product repairs, maintenance services, and technical support services. We are focusing on identifying hardware systems support processes that are intended to proactively identify and solve quality issues and to increase the amount of new hardware systems support contracts sold in connection with the sales of our hardware systems products. Hardware systems support contracts are generally priced as a percentage of the net hardware systems products fees. Our hardware systems support revenues represented 3% of our total revenues in fiscal 2010.

Services Business

Our services business consists of consulting, On Demand and education. As a result of our acquisition of Sun, we expanded and enhanced our customer base and services offerings.

Consulting

Oracle Consulting is designed to help our customers more successfully deploy our products. Our consulting services include: enterprise architecture design and implementation; business/IT strategy alignment; business process simplification; solution integration; and product implementation, enhancements, and upgrades. These services help our customers achieve their business goals, manage their total cost of ownership and reduce the risk associated with their product deployment. Oracle Consulting employs consulting professionals globally to engage our customers directly, as well as to provide specialized expertise to our global systems integrator partners. Oracle Consulting utilizes a global blended delivery model to achieve economies of scale for our customers. This global delivery model consists of onsite consultants within the customer's local geography as well as consultants in our global delivery and solution centers. Consulting revenues represented 10%, 14%, and 15% of total revenues in fiscal 2010, 2009 and 2008, respectively.

On Demand

On Demand includes our Oracle On Demand and Advanced Customer Services offerings. Oracle On Demand is designed to provide multi-featured software and hardware management and maintenance services for customers that are delivered at our data center facilities, select partner data centers or physically on-site at customer facilities. Advanced Customer Services consists of solution lifecycle management services, database and application management services, industry-specific solution support centers, and remote and on-site expert services. As a result of our acquisition of Sun, we increased the volume and breadth of our Advanced Customer Services offerings, including onsite and remote operations management, hardware systems expert services, and packaged offerings that support the installation and optimization of Sun systems. We believe that our On Demand offerings provide our customers flexibility in how they manage their IT environments and an additional opportunity to lower their total cost of ownership and can therefore provide us with a competitive advantage. On Demand revenues represented 3% of total revenues in each of fiscal 2010, 2009 and 2008.

Education

We provide training to customers, partners and employees as a part of our mission of accelerating the adoption and use of our software and hardware products and to create opportunities to grow our product revenues. Our training is provided through a variety of formats, including instructor-led classes at our education centers, live

virtual training, self-paced online training, training via CD-ROM, private events and custom training. Our live virtual class offerings allow students anywhere in the world to receive real-time, interactive training online. In addition, we also offer a certification program certifying database administrators, developers, implementers, consultants and architects. Education revenues represented 1% of our total revenues in fiscal 2010 and 2% of total revenues in fiscal 2009 and 2008, respectively.

Marketing and Sales

We directly market and sell our products and services primarily through our subsidiary sales and service organizations to businesses of many sizes and in many industries, government agencies and educational institutions. We also market, and sell our products through indirect channels. No single customer accounted for 10% or more of our total revenues in fiscal 2010, 2009 or 2008.

In the United States, our sales and service employees are based in our headquarters and in field offices throughout the country. Outside the United States, our international subsidiaries license and support our products in their local countries as well as within other foreign countries where we do not operate through a direct sales subsidiary. Our geographic coverage allows us to draw on business and technical expertise from a global workforce, provides stability to our operations and revenue streams to offset geography-specific economic trends and offers us an opportunity to take advantage of new markets for our products. A summary of our domestic and international revenues and long-lived assets is set forth in Note 16 of Notes to Consolidated Financial Statements.

We also market our products worldwide through indirect channels. The companies that comprise our indirect channel network are members of the Oracle Partner Network. The Oracle Partner Network is a global program that manages our business relationships with a large, broad-based network of companies, including independent software and hardware vendors, system integrators and resellers who deliver innovative solutions and services based upon our products. By offering our partners access to our premier products, educational information, technical services, marketing and sales support, the Oracle Partner Network program extends our market reach by providing our partners with the resources they need to be successful in delivering solutions to customers globally. Prior to our acquisition of Sun, the majority of Sun's hardware systems products were sold through indirect channels. Although we plan to continue to sell our hardware products through indirect channels, including independent distributors and value added resellers, we have begun enhancing direct sales coverage for our hardware systems products and intend that our direct sales force will sell proportionately more of our hardware systems products in the future than they do currently.

Seasonality and Cyclicality

Our quarterly results reflect distinct seasonality in the sale of our products and services. Our total revenues and operating margins are typically highest in our fourth fiscal quarter and lowest in our first fiscal quarter. General economic conditions also have an impact on our business and financial results. The markets in which we sell our products and services have, at times, experienced weak economic conditions that have negatively affected our revenues. See "Selected Quarterly Financial Data" in Item 7 of this Annual Report for a more complete description of the seasonality and cyclicality of our revenues, expenses and margins.

Competition

We face intense competition in all aspects of our business. The nature of the IT industry creates a competitive landscape that is constantly evolving as firms emerge, expand or are acquired, as technology evolves and as customer demands and competitive pressures otherwise change.

Our customers are demanding less complexity and lower total cost in the implementation, sourcing, integration and ongoing maintenance of their enterprise software and hardware systems, which has led increasingly to our product offerings being viewed as a "stack" of software and hardware designed to work together in a standards-compliant environment—from "Applications to Disk." Our enterprise software and hardware offerings compete directly with some offerings from the most competitive companies in the world, including Microsoft Corporation (Microsoft), IBM Corporation (IBM), Hewlett Packard Company (HP), SAP AG, and Intel, as well as many

others. In addition, the low barriers to entry in many of our market segments regularly introduce new technologies and new and growing competitors to challenge our offerings. Our competitors range from companies offering broad IT solutions across many of our lines of business to vendors providing point solutions, or offerings focused on a specific functionality, product area or industry. In addition, as we expand into new market segments, we will face increased competition as we will compete with existing competitors, as well as firms that may be partners in other areas of our business and other firms with whom we have not previously competed. Moreover, we or our competitors may take certain strategic actions—including acquisitions, partnerships and joint ventures, or repositioning of product lines—which invite even greater competition in one or more product categories.

Key competitive factors in each of the segments in which we currently compete and may compete in the future include: total cost of ownership, performance, scalability, reliability, security, functionality, efficiency, ease of management and quality of technical support. Our product sales (and the relative strength of our products versus those of our competitors) are also directly and indirectly affected by the following, among other things:

- the broader "platform" competition between our industry standard Java technology platform and the .NET programming environment of Microsoft;
- operating system competition among, primarily, our Solaris operating system, Microsoft's Windows Server, Unix (including HP-UX from HP and AIX from IBM) and Linux;
- the adoption of commodity servers and microprocessors;
- the adoption of open source alternatives to commercial software by enterprise software customers;
- the adoption of Software-as-a-Service (SaaS), hosted or "cloud" software offerings;
- products, features and functionality developed internally by customers and their IT staff;
- products, features or functionality customized and implemented for customers by consultants, systems integrators or other third parties; and
- attractiveness of offerings from business processing outsourcers.

For more information about the competitive risks we face, refer to Item 1A. "Risk Factors."

Manufacturing

To produce our hardware systems products, we rely on both our internal manufacturing operations as well as third party manufacturing partners. Our internal manufacturing operations consist primarily of final assembly, test and quality control of our enterprise and data center servers and storage systems. For all other manufacturing, we rely on third party manufacturing partners. We distribute most of our hardware systems products either from our facilities or partner facilities. One of our main goals is to reduce costs by simplifying our manufacturing processes through increased standardization of components across product types, through a reduction of the number of assembly and distribution centers we rely on and through our transition to a "build-to-order" process in which products are built only after customers have placed firm orders. Production of our hardware products requires that we purchase materials, supplies, product subassemblies and full assemblies from a number of vendors. For most of our hardware products, we have existing alternate sources of supply or such sources are readily available. However, we do rely on sole sources for certain of our hardware products. For example, we have a long-standing relationship with Fujitsu Limited for the development, manufacturing and marketing of certain of our SPARC server components and products. Refer to "Risk Factors" included in Item 1A. of this Annual Report for additional discussion of the challenges we encounter with respect to the sources and availability of supplies for our products and the related risks to our business.

Research and Development

We develop the substantial majority of our products internally. In addition, we have extended our product offerings and intellectual property through acquisitions of businesses and technologies. We also purchase or license intellectual property rights in certain circumstances. Internal development allows us to maintain technical control over the design and development of our products. We have a number of United States and foreign patents

and pending applications that relate to various aspects of our products and technology. While we believe that our patents have value, no single patent is essential to us or to any of our principal business segments. Research and development expenditures were $3.3 billion, $2.8 billion and $2.7 billion, in fiscal 2010, 2009 and 2008, respectively, or 12% of total revenues in each of the aforementioned fiscal years. Rapid technological advances in hardware and software development, evolving standards in computer hardware and software technology, changing customer needs and frequent new product introductions and enhancements characterize the software and hardware markets in which we compete. We plan on continuing to dedicate a significant amount of resources to research and development efforts to maintain and improve our current product offerings.

Employees

As of May 31, 2010, we employed approximately 105,000 full-time employees, including approximately 24,000 in sales and marketing, approximately 9,000 in software license updates and product support, approximately 1,000 in the manufacturing of our hardware systems products, approximately 5,000 in hardware systems support, approximately 26,000 in services, approximately 28,000 in research and development and approximately 12,000 in general and administrative positions. Of these employees, approximately 39,000 were located in the United States and approximately 66,000 were employed internationally. None of our employees in the United States is represented by a labor union; however, in certain foreign subsidiaries workers' councils represent our employees.

Available Information

Our Annual Report on Form 10-K, Quarterly Reports on Form 10-Q, Current Reports on Form 8-K and amendments to reports filed pursuant to Sections 13(a) and 15(d) of the Securities Exchange Act of 1934, as amended, are available, free of charge, on our Investor Relations web site at www.oracle.com/investor as soon as reasonably practicable after we electronically file such material with, or furnish it to, the SEC. The information posted on our web site is not incorporated into this Annual Report.

Executive Officers of the Registrant

Our executive officers are listed below.

Name	Office(s)
Lawrence J. Ellison	Chief Executive Officer and Director
Jeffrey O. Henley	Chairman of the Board of Directors
Safra A. Catz	President and Director
Charles E. Phillips, Jr.	President and Director
Keith G. Block	Executive Vice President, North America Sales and Consulting
Jeff Epstein	Executive Vice President and Chief Financial Officer
John Fowler	Executive Vice President, Systems
Thomas Kurian	Executive Vice President, Product Development
Loic Le Guisquet	Executive Vice President, Europe, Middle East and Africa (EMEA) Sales and Consulting
Luiz Meisler	Executive Vice President, Latin America Sales and Consulting
Juergen Rottler	Executive Vice President, Oracle Customer Services
Charles A. Rozwat	Executive Vice President
Mike Splain	Executive Vice President, Microelectronics Group
Derek H. Williams	Executive Vice President, Japan Sales and Consulting
Dorian E. Daley	Senior Vice President, General Counsel and Secretary
William Corey West	Senior Vice President, Corporate Controller and Chief Accounting Officer

Mr. Ellison, 65, has been Chief Executive Officer and a Director since he founded Oracle in June 1977. He served as Chairman of the Board from May 1995 to January 2004.

Mr. Henley, 65, has served as the Chairman of the Board since January 2004 and as a Director since June 1995. He served as an Executive Vice President and Chief Financial Officer from March 1991 to July 2004.

Ms. Catz, 48, has been a President since January 2004 and has served as a Director since October 2001. She was Chief Financial Officer from November 2005 until September 2008 and Interim Chief Financial Officer from April 2005 until July 2005. She served as an Executive Vice President from November 1999 to January 2004 and Senior Vice President from April 1999 to October 1999. She also serves as a director of HSBC Holdings plc.

Mr. Phillips, 51, has been a President and has served as a Director since January 2004. He served as Executive Vice President, Strategy, Partnerships, and Business Development, from May 2003 to January 2004. He also serves as a director of Viacom, Inc.

Mr. Block, 49, has been Executive Vice President, North America Sales and Consulting since September 2002. He served as Executive Vice President of North America Consulting from February to September of 2002. He served as Senior Vice President of North America Commercial Consulting and Global Service Lines from June 1999 until January 2002. He also held various other positions with us since joining Oracle in 1985.

Mr. Epstein, 53, has been Executive Vice President and Chief Financial Officer since September 2008. Prior to joining us, he served as Executive Vice President and Chief Financial Officer of Oberon Media, Inc., a privately held internet game technology provider and publisher from April 2007 to June 2008. From June 2005 until its sale in March 2007, Mr. Epstein was Executive Vice President and Chief Financial Officer of ADVO, Inc., a direct mail media company. Mr. Epstein was a member of the Board of Directors of Revonet, Inc., a business-to-business marketing and database company, from January 2004 to December 2005, Chairman of the Board from December 2004 to December 2005 and the Acting President and Chief Executive Officer from June 2004 through December 2004. Mr. Epstein also serves as a director of priceline.com Incorporated and serves on the Audit and Compliance Committee of the Stanford University Hospital.

Mr. Fowler, 49, has been Executive Vice President, Systems since February 2010. Prior to joining us, Mr. Fowler served as Sun Microsystems, Inc.'s Executive Vice President, Systems Group from May 2006 to February 2010, as Executive Vice President, Network Systems Group from May 2004 to May 2006, as Chief Technology Officer, Software Group from July 2002 to May 2004 and Director, Corporate Development from July 2000 to July 2002.

Mr. Kurian, 43, has been Executive Vice President, Product Development since July 2009. He served as Senior Vice President of Development from February 2001 until July 2009. Mr. Kurian worked in Oracle Server Technologies as Vice President of Development from March 1999 until February 2001. He also held various other positions with us since joining Oracle in 1996.

Mr. Le Guisquet, 48, has been Executive Vice President, Europe, Middle East and Africa (EMEA) Sales and Consulting since December 2008. He served as Senior Vice President, Europe Applications from June 2006 until November 2008. He served as Senior Vice President, Oracle Customer Relationship Management EMEA from January 2006 until June 2006. He served as Senior Vice President, EMEA Consulting from August 2003 until January 2006. He also held various other EMEA regional executive positions with us since January 1990.

Mr. Meisler, 57, has been Executive Vice President, Latin America Sales and Consulting since July 2008. He served as Senior Vice President, Latin America Sales and Consulting from December 2001 to July 2008; as Vice President, Latin America Sales and Consulting from June 2001 to December 2001; and as Managing Director of Oracle Brazil from January 2000 to May 2001. He served as Vice President, Latin America Consulting from June 1999 to January 2000 and as Vice President, Oracle Brazil Consulting from March 1998 to May 1999.

Mr. Rottler, 43, has been Executive Vice President, Oracle Customer Services since September 2006. He was Executive Vice President, Oracle Support and Oracle On Demand, from January 2005 to September 2006 and was Executive Vice President, Oracle On Demand, from September 2004 to January 2005.

Mr. Rozwat, 62, has served as Executive Vice President, Product Development from October 2007 to July 2009, as Executive Vice President, Server Technologies from November 1999 to October 2007 and as Senior Vice President, Database Server from December 1996 to October 1999. He served as Vice President of Development from December 1994 to November 1996.

Mr. Splain, 53, has been Executive Vice President, Microelectronics Group since February 2010. Prior to joining us, Mr. Splain served as Sun's Executive Vice President, Microelectronics from April 2008 to February 2010, as

Chief Engineer from June 2006 to February 2010, as Chief Technology Officer, Systems Group from June 2006 to April 2008, Chief Technology Officer, Scalable Systems from March 2004 to June 2006, and Chief Technology Officer, Processor and Network Products from June 2002 to March 2004.

Mr. Williams, 65, has been Executive Vice President, Japan Sales and Consulting since June 2008 and was Executive Vice President, Asia Pacific Sales and Consulting from October 2000 to May 2008. He served as Senior Vice President, Asia Pacific from July 1993 to October 2000 and as Vice President, Asia Pacific from April 1991 to July 1993. He joined Oracle United Kingdom in October 1988 and served as Regional Director, Strategic Accounts from October 1988 to April 1991.

Ms. Daley, 51, has been Senior Vice President, General Counsel and Secretary since October 2007. She served as Vice President, Legal, Associate General Counsel and Assistant Secretary from June 2004 to October 2007, as Associate General Counsel and Assistant Secretary from October 2001 to June 2004, and as Associate General Counsel from February 2001 to October 2001. She joined Oracle's Legal Department in 1992.

Mr. West, 48, has been Senior Vice President, Corporate Controller and Chief Accounting Officer since February 2008 and was Vice President, Corporate Controller and Chief Accounting Officer from April 2007 to February 2008. Prior to joining us, he served as Intuit Inc.'s Director of Accounting from August 2005 to March 2007, as The Gap, Inc.'s Assistant Controller from April 2005 to August 2005, and as Vice President, Finance, at Cadence Design Systems, Inc.'s product business from June 2001 to April 2005. He also spent 14 years with Arthur Andersen LLP, most recently as a partner.

Item 1A. Risk Factors

We operate in a rapidly changing economic and technological environment that presents numerous risks, many of which are driven by factors that we cannot control or predict. The following discussion, as well as our "Critical Accounting Policies and Estimates" discussion in Management's Discussion and Analysis of Financial Condition and Results of Operations (Item 7), highlights some of these risks. The risks described below are not exhaustive and you should carefully consider these risks and uncertainties before investing in our securities.

Economic, political and market conditions, including the recent recession and global economic crisis, can adversely affect our business, results of operations and financial condition, including our revenue growth and profitability, which in turn could adversely affect our stock price. Our business is influenced by a range of factors that are beyond our control and that we have no comparative advantage in forecasting. These include:

- general economic and business conditions;
- currency exchange rate fluctuations;
- the overall demand for enterprise software, hardware systems and services;
- governmental budgetary constraints or shifts in government spending priorities; and
- general political developments.

The recent recession and global economic crisis caused a general tightening in the credit markets, lower levels of liquidity, increases in the rates of default and bankruptcy, and extreme volatility in credit, equity and fixed income markets. These macroeconomic developments negatively affected, and could continue to negatively affect, our business, operating results or financial condition which, in turn, could adversely affect our stock price. A general weakening of, and related declining corporate confidence in, the global economy or the curtailment in government or corporate spending could cause current or potential customers to reduce their IT budgets or be unable to fund software, hardware systems or services purchases, which could cause customers to delay, decrease or cancel purchases of our products and services or cause customers not to pay us or to delay paying us for previously purchased products and services.

In some financial markets, institutions may decrease or discontinue their purchase of the long-term customer financing contracts that we have traditionally sold on a non-recourse basis. As a result, we may hold more of these contracts ourselves or require more customers to purchase our products and services on a cash basis.

In addition, terrorist attacks around the world, the wars in Afghanistan and Iraq and the potential for other hostilities in various parts of the world, potential public health crises and natural disasters continue to contribute to a climate of economic and political uncertainty that could adversely affect our results of operations and financial condition, including our revenue growth and profitability. These factors generally have the strongest effect on our sales of new software licenses, hardware systems products, hardware systems support and related services and, to a lesser extent, also may affect our renewal rates for software license updates and product support.

We may fail to achieve our financial forecasts due to inaccurate sales forecasts or other factors. Our revenues, and particularly our new software license revenues and hardware systems products revenues, are difficult to forecast, and, as a result, our quarterly operating results can fluctuate substantially. Our limited experience with managing our new hardware business and forecasting its future financial results creates additional challenges with our forecasting processes.

We use a "pipeline" system, a common industry practice, to forecast sales and trends in our business. Our sales personnel monitor the status of all proposals and estimate when a customer will make a purchase decision and the dollar amount of the sale. These estimates are aggregated periodically to generate a sales pipeline. Our pipeline estimates can prove to be unreliable both in a particular quarter and over a longer period of time, in part because the "conversion rate" or "closure rate" of the pipeline into contracts can be very difficult to estimate. A contraction in the conversion rate, or in the pipeline itself, could cause us to plan or budget incorrectly and adversely affect our business or results of operations. In particular, a slowdown in IT spending or economic conditions generally can unexpectedly reduce the conversion rate in particular periods as purchasing decisions are delayed, reduced in amount or cancelled. The conversion rate can also be affected by the tendency of some of our customers to wait until the end of a fiscal period in the hope of obtaining more favorable terms, which can also impede our ability to negotiate and execute these contracts in a timely manner. In addition, for newly acquired companies, we have limited ability to predict how their pipelines will convert into sales or revenues for at least one or two quarters following the acquisition, and potentially longer with respect to our acquisition of Sun. Conversion rates post-acquisition may be quite different from the acquired companies' historical conversion rates. Differences in conversion rates can also be affected by changes in our business practices that we implement with our newly acquired companies that may affect customer behavior.

A substantial portion of our new software license revenue contracts and hardware systems products contracts is completed in the latter part of a quarter and a significant percentage of these are large orders. Because a significant portion of our cost structure is largely fixed in the short term, revenue shortfalls tend to have a disproportionately negative impact on our profitability. The number of large new software license transactions, and to a lesser extent hardware systems products transactions, also increases the risk of fluctuations in our quarterly results because a delay in even a small number of these transactions could cause our quarterly revenues and profitability to fall significantly short of our predictions.

We may not achieve our financial forecasts with respect to our acquisition of Sun or our entrance into a new hardware systems business, or the achievement of such forecasts may take longer than expected. Our profitability could decline if we do not manage the risks associated with our acquisition and integration of Sun. The acquisition and ongoing integration of Sun into Oracle may adversely affect our profitability if we do not manage the associated risks. We may not achieve the anticipated synergies, cost savings, customer and partner advantages and benefits or realize our estimated revenue, profit or other financial projections in a timely manner or at all due to a number of factors, including:

- Prior to the acquisition of Sun, we had limited or no direct experience in managing certain aspects of a hardware systems business. While we plan to retain a large number of Sun engineers, salespeople and other employees with experience in managing this business, our inexperience or the unplanned departures of some important employees could adversely impact our ability to successfully manage our hardware systems business, which could adversely impact our ability to realize the forecasts for our hardware systems business and its results of operations.

- Our plans to reduce the costs associated with managing our hardware systems business, through expected efficiencies from, among other things, (i) changes in supply chain processes or in customer services and support or (ii) eliminations of redundancies in personnel, facilities or other services, may not be realized or completed within the expected time frame, if at all, or the anticipated cost savings may be less than what we forecast.

- We may not be able to increase sales of hardware systems support contracts, which could result in lower hardware systems or hardware systems support revenues and profitability, or slower than expected growth of such revenues and profitability.

- Sun's hardware systems business historically has had higher expenses as a percentage of revenues, and thus has been less profitable, than our standalone software business. Upon completion of our acquisition of Sun, we have reported, and may continue to report, lower operating margins as a percentage of revenues, and our profit margin levels prior to our acquisition of Sun may not be sustainable.

- We face a greater risk of potential write-downs and impairments of inventory, higher warranty expenses than we had historically encountered in our existing software and services businesses, and higher amortization from, and potential impairment of, intangible assets associated with our hardware systems business. Any of these items could result in material charges and adversely affect our operating results.

- Our senior management's attention will be diverted from our software business to our hardware systems business, which may be disruptive to our software business or the overall management of Oracle.

Our success depends upon our ability to develop new products and services, integrate acquired products and services and enhance our existing products and services. Rapid technological advances and evolving standards in computer hardware and software development and communications infrastructure, changing and increasingly sophisticated customer needs and frequent new product introductions and enhancements characterize the enterprise software and hardware systems markets in which we compete. If we are unable to develop new or sufficiently differentiated products and services, or to enhance and improve our products and support services in a timely manner or to position and/or price our products and services to meet market demand, customers may not buy new software licenses or hardware systems products or purchase or renew software license updates and product support or hardware systems support contracts. Renewals of these support contracts are important to the growth of our business. In addition, IT standards from both consortia and formal standards-setting forums as well as de facto marketplace standards are rapidly evolving. We cannot provide any assurance that the standards on which we choose to develop new products will allow us to compete effectively for business opportunities in emerging areas.

We are currently building and testing the next generation of our applications software offerings, which are being designed to unify the best-of-business functional capabilities from all of our applications on a modern Internet-based middleware technology foundation. We have also recently designed and built the Sun Oracle Database Machine, a fast database warehousing machine that runs online transaction processing applications. If we do not continue to develop and release these or other new or enhanced products and services within the anticipated time frames, if there is a delay in market acceptance of a new, enhanced or acquired product line or service, if we do not timely optimize complementary product lines and services or if we fail to adequately integrate, support or enhance acquired product lines or services, our business may be adversely affected.

Our strategy of transitioning from Sun's indirect sales model to our mixed direct and indirect sales model may not succeed and could result in lower hardware systems revenues or profits. Disruptions to our software indirect sales channel could affect our future operating results. Prior to our acquisition of Sun, the majority of Sun's hardware systems products were sold through indirect channels. Although we plan to continue to sell our hardware products through indirect channels, including independent distributors and value added resellers, we have begun enhancing direct sales coverage for our hardware products and intend that our direct sales force will sell a larger portion of our hardware products in the future than they do now. These direct sales efforts, however, may not be successful. Our relationships with some of our channel partners may deteriorate because we

are reducing our reliance on some of these partners for sales of our hardware products, are modifying our approach and timing to the manufacturing of our products and are altering certain of Sun's legacy business practices with these channel partners, which could result in reduced demand from the channel partners or certain customer segments serviced by these channel partners. Some hardware systems revenues from channel partners may not be replaced by revenues generated from our own sales personnel or be replaced as quickly as we expect. In addition, we may not be able to hire qualified hardware systems salespeople, sales consultants and other personnel for our direct sales model at the rate or in the numbers we need to generate the hardware systems revenues and profit margins we have projected for future periods. Even if we can meet our hiring needs, these salespeople may not be able to achieve our sales forecasts for our hardware systems business. If we experience any of these risks, our hardware systems revenues or profits may decline.

Our software indirect channel network is comprised primarily of resellers, system integrators/implementers, consultants, education providers, internet service providers, network integrators and independent software vendors. Our relationships with these channel participants are important elements of our software marketing and sales efforts. Our financial results could be adversely affected if our contracts with channel participants were terminated, if our relationships with channel participants were to deteriorate, if any of our competitors enter into strategic relationships with or acquire a significant channel participant or if the financial condition of our channel participants were to weaken. There can be no assurance that we will be successful in maintaining, expanding or developing our relationships with channel participants. If we are not successful, we may lose sales opportunities, customers and revenues.

If we are unable to compete effectively with existing or new hardware systems or software competitors, the results of operations and prospects for our business could be harmed through fewer customer orders, reduced pricing, lower revenues or lower profits. Our hardware systems business will compete with, among others, (i) systems manufacturers and resellers of systems based on our own microprocessors and operating systems and those of our competitors, (ii) microprocessor/chip manufacturers and (iii) providers of storage products. Our entrance into the hardware systems business may also cause us to compete with companies who historically have been our partners. These competitors in most cases have more experience than we do in managing a hardware business. A large portion of our hardware products are based on our SPARC microprocessor and Solaris operating system platform, which has a smaller installed base than certain of our competitors' platforms and which may make it difficult for us to win new customers that have already made significant investments in our competitors' platforms. Certain of these competitors also compete very aggressively on price. Furthermore, Sun experienced increased competition during the pendency of our acquisition of Sun. A loss in our competitive position could result in lower revenues or profitability, which could adversely impact our ability to realize the revenue and profitability forecasts for our hardware systems business.

Many vendors develop and market databases, middleware products, application development tools, business applications, collaboration products and business intelligence products that compete with our software offerings. In addition, several companies offer business process outsourcing (BPO) and Software-as-a-Service (SaaS) as competitive alternatives to buying software and hardware, and customer interest in BPO and SaaS solutions is increasing. Some of these competitors have greater financial or technical resources than we do. Our competitors that offer business applications and middleware products may influence a customer's purchasing decision for the underlying database in an effort to persuade potential customers not to acquire our products. We could lose customers if our competitors introduce new competitive products, add new functionality, acquire competitive products, reduce prices or form strategic alliances with other companies. Vendors that offer BPO or SaaS solutions may persuade our customers not to purchase our products. We may also face increasing competition from open source software initiatives, in which competitors may provide software and intellectual property for free. Existing or new competitors could gain sales opportunities or customers at our expense.

Our hardware systems offerings are complex products. If we cannot successfully manage the required processes to meet customer requirements and demand on a timely basis, the results of our hardware systems business will suffer. Designing, developing, manufacturing and introducing new hardware products are complicated processes. The development process is uncertain and requires a high level of innovation from both systems hardware and software product designers and engineers and the suppliers of the components used in these products. The development process is also lengthy and costly. Once a new hardware product is developed, we face several

challenges in the manufacturing process. We must be able to forecast customer demand and manufacture new hardware products in sufficient volumes to meet this demand and do so in a cost effective manner. As we continue to transition to a "build-to-order" manufacturing model where our hardware products are not fully assembled until after customers place orders, we may from time to time experience delays in delivering our hardware products to customers in a timely manner. These delays could cause our customers to purchase hardware products and services from our competitors. We must also manage new hardware product introductions and transitions to minimize the impact of customer delayed purchases of existing hardware products in anticipation of new hardware product releases. Because the design and manufacturing process for components is also very complicated, it is possible that we could experience design or manufacturing flaws. These design or manufacturing flaws could delay or prevent the production of the components for which we have previously committed to pay or need to fulfill orders from customers. These types of component flaws could also prevent the production of our hardware products or cause our hardware products to be returned, recalled or rejected resulting in lost revenues, increases in warranty costs, damage to our reputation, penalties and litigation.

Acquisitions present many risks, and we may not realize the financial and strategic goals that were contemplated at the time of a transaction. In recent years, we have invested billions of dollars to acquire a number of companies, products, services and technologies. An active acquisition program is an important element of our overall corporate strategy, and we expect to continue to make similar acquisitions in the future. Risks we may face in connection with our acquisition program include:

- our ongoing business may be disrupted and our management's attention may be diverted by acquisition, transition or integration activities;
- an acquisition may not further our business strategy as we expected, we may not integrate an acquired company or technology as successfully as we expected or we may overpay for, or otherwise not realize the expected return on, our investments, which could adversely affect our business or operating results;
- we may have difficulties (i) managing an acquired company's technologies or lines of business or (ii) entering new markets where we have no or limited direct prior experience or where competitors may have stronger market positions;
- our operating results or financial condition may be adversely impacted by claims or liabilities that we assume from an acquired company or technology or that are otherwise related to an acquisition, including claims from government agencies, terminated employees, current or former customers, former stockholders or other third parties; pre-existing contractual relationships of an acquired company that we would not have otherwise entered into, the termination or modification of which may be costly or disruptive to our business; unfavorable revenue recognition or other accounting treatment as a result of an acquired company's practices; and intellectual property claims or disputes;
- we may fail to identify or assess the magnitude of certain liabilities, shortcomings or other circumstances prior to acquiring a company or technology, which could result in unexpected litigation or regulatory exposure, unfavorable accounting treatment, unexpected increases in taxes due, a loss of anticipated tax benefits or other adverse effects on our business, operating results or financial condition;
- we may not realize the anticipated increase in our revenues from an acquisition for a number of reasons, including if a larger than predicted number of customers decline to renew software license updates and product support contracts and hardware systems support contracts, if we are unable to sell the acquired products to our customer base or if contract models of an acquired company do not allow us to recognize revenues on a timely basis;
- we may have difficulty incorporating acquired technologies or products with our existing product lines and maintaining uniform standards, architecture, controls, procedures and policies;
- we may have multiple product lines as a result of our acquisitions that are offered, priced and supported differently, which could cause customer confusion and delays;
- we may have higher than anticipated costs in continuing support and development of acquired products, in general and administrative functions that support new business models, or in compliance with associated regulations that are more complicated than we had anticipated;

- we may be unable to obtain timely approvals from, or may otherwise have certain limitations, restrictions, penalties or other sanctions imposed on us by, worker councils or similar bodies under applicable employment laws as a result of an acquisition, which could adversely affect our integration plans in certain jurisdictions;

- we may be unable to obtain required approvals from governmental authorities under competition and antitrust laws on a timely basis, if it all, which could, among other things, delay or prevent us from completing a transaction, otherwise restrict our ability to realize the expected financial or strategic goals of an acquisition or have other adverse effects on our current business and operations;

- our use of cash to pay for acquisitions may limit other potential uses of our cash, including stock repurchases, dividend payments and retirement of outstanding indebtedness;

- we may significantly increase our interest expense, leverage and debt service requirements if we incur additional debt to pay for an acquisition and we may have to delay or not proceed with a substantial acquisition if we cannot obtain the necessary funding to complete the acquisition in a timely manner or on favorable terms;

- we may experience additional or unexpected changes in how we are required to account for our acquisitions pursuant to U.S. generally accepted accounting principles, including arrangements that we assume from an acquisition; and

- to the extent that we issue a significant amount of equity securities in connection with future acquisitions, existing stockholders may be diluted and earnings per share may decrease.

The occurrence of any of these risks could have a material adverse effect on our business, results of operations, financial condition or cash flows, particularly in the case of a larger acquisition or several concurrent acquisitions.

Our international sales and operations subject us to additional risks that can adversely affect our operating results. We derive a substantial portion of our revenues from, and have significant operations, outside of the United States. Our international operations include software and hardware systems development, manufacturing, sales, customer support, consulting, On Demand and shared administrative service centers.

Compliance with international and U.S. laws and regulations that apply to our international operations increases our cost of doing business in foreign jurisdictions. These laws and regulations include U.S. laws such as the Foreign Corrupt Practices Act, and local laws which also prohibit corrupt payments to governmental officials, data privacy requirements, labor relations laws, tax laws, anti-competition regulations, import and trade restrictions, and export requirements. Violations of these laws and regulations could result in fines, criminal sanctions against us, our officers or our employees, and prohibitions on the conduct of our business. Any such violations could result in prohibitions on our ability to offer our products and services in one or more countries, could delay or prevent potential acquisitions, and could also materially damage our reputation, our brand, our international expansion efforts, our ability to attract and retain employees, our business and our operating results. Our success depends, in part, on our ability to anticipate these risks and manage these difficulties. We monitor our international operations and investigate allegations of improprieties relating to transactions and the way in which such transactions are recorded. Where circumstances warrant, we provide information and report our findings to government authorities, but no assurance can be given that action will not be taken by such authorities.

We are also subject to a variety of other risks and challenges in managing an organization operating in various countries, including those related to:

- general economic conditions in each country or region;

- fluctuations in currency exchange rates and related impacts to our operating results;

- regulatory changes;

- political unrest, terrorism and the potential for other hostilities;

- public health risks, particularly in areas in which we have significant operations;

- longer payment cycles and difficulties in collecting accounts receivable;
- overlapping tax regimes;
- our ability to repatriate funds held by our foreign subsidiaries to the United States at favorable tax rates;
- difficulties in transferring funds from or converting currencies in certain countries; and
- reduced protection for intellectual property rights in some countries.

As a result of our entry into a new hardware systems business, the volume and complexity of laws and regulations that we are subject to have increased.

As the majority shareholder of Oracle Financial Services Software Limited, a publicly traded Indian software company focused on the banking industry, we are faced with several additional risks, including being subject to local securities regulations and being unable to exert full control or obtain financial and other information on a timely basis.

We may experience foreign currency gains and losses. We conduct a significant number of transactions in currencies other than the U.S. Dollar. Changes in the value of major foreign currencies, particularly the Euro, Japanese Yen and British Pound relative to the U.S. Dollar can significantly affect revenues and our operating results. Generally, our revenues and operating results are adversely affected when the dollar strengthens relative to other currencies and are positively affected when the dollar weakens. For example, our revenues and operating results in fiscal 2009 were unfavorably affected by the strengthening of the U.S. Dollar relative to most other major foreign currencies.

In addition, we incur foreign currency transaction gains and losses, primarily related to sublicense fees and other intercompany agreements among us and our subsidiaries and distributors that we expect to cash settle in the near term, that are charged against earnings in the period incurred. We have a program that primarily utilizes foreign currency forward contracts to offset the risks associated with these foreign currency exposures, which we may suspend from time to time. This program was active for the majority of fiscal 2010 and was suspended during our fourth quarter of fiscal 2010. When the program is active, we enter into foreign currency forward contracts so that increases or decreases in our foreign currency exposures are offset by gains or losses on the foreign currency forward contracts in order to mitigate the risks and volatility associated with our foreign currency transaction gains or losses. As a large portion of our consolidated operations are international, we could experience additional foreign currency volatility in the future, the amounts and timing of which are variable. We will continue to experience foreign currency gains and losses in certain instances when we suspend our foreign currency forward contract program or where it is not possible or cost effective to hedge our foreign currency exposures. For example, if overall foreign currency exchange rates weakened subsequent to May 31, 2010, we believe we would recognize foreign currency losses related to cross-currency exposures that our foreign currency forward contracts program, when active, may be able to mitigate. Our ultimate realized loss or gain with respect to currency fluctuations will generally depend on the size and type of cross-currency exposures that we enter into, the currency exchange rates associated with these exposures and changes in those rates, whether we have entered into foreign currency forward contracts to offset these exposures and other factors. All of these factors could materially impact our results of operations, financial position and cash flows, the timing of which is variable and generally outside of our control. For example, during fiscal 2010, we incurred foreign currency losses associated with our Venezuelan subsidiary due to it being designated as operating in a highly inflationary economy and due to the subsequent devaluation of the Venezuelan Bolivar relative to the U.S. Dollar.

The future operating results of our hardware systems business will depend on our ability to manage our component inventory to meet the demands of our hardware systems customers and to avoid component inventory write-downs. Sun's hardware systems business has historically depended upon suppliers to design, develop, manufacture and deliver on a timely basis the necessary components for its hardware products. While many of the components historically purchased have been standard, some components (standard or otherwise) have required long lead times to manufacture and deliver. Furthermore, there are some components that can only be purchased from a single vendor due to price, quality, technology or other business constraints. At times we may be unable to purchase these items from the respective single vendors on acceptable terms or may experience significant delays or

quality issues in the delivery of necessary parts or components from a particular vendor. If we had to find a new supplier for these parts and components, hardware product shipments could be delayed. We could also experience fluctuations in component prices which, if unanticipated, could negatively impact our hardware systems business cost structure. These factors may make it difficult for us to plan and procure appropriate component inventory levels in a timely fashion to meet customer demand for our hardware products. Therefore we may experience component inventory shortages which may result in production delays or customers choosing to purchase fewer hardware products from us or systems products from our competitors. Sun historically has negotiated supply commitments with vendors early in the manufacturing process to ensure they had sufficient components for its hardware systems products to meet anticipated customer demand, and we initially expect to do the same. We must also manage our levels of older component inventories used in our hardware products to minimize inventory write-offs or write-downs. If we have excess inventory, it may be necessary to write-down the inventory, which would adversely affect our operating results. If one or more of the risks described above occurs, our hardware systems business and related operating results could be materially and adversely affected.

We expect to continue to depend on third party manufacturers to build certain hardware systems products and are susceptible to manufacturing delays that could prevent us from shipping customer orders on time, if at all, and may result in the loss of sales and customers. We outsource the manufacturing and assembly of certain of our hardware products to a variety of manufacturing companies, many of which are located outside the United States. Our reliance on third-party manufacturers reduces our control over the manufacturing process, exposing us to risks, including reduced control over quality assurance, product costs and product supply as well as the political and economic uncertainties of the international locations where certain of these third-party manufacturers have facilities and operations. Any manufacturing disruption by our third-party manufacturers could impair our ability to fulfill orders for these hardware systems products. If we are unable to manage our relationships with these third-party manufacturers effectively, or if these third-party manufacturers experience delays, disruptions, capacity constraints, regulatory issues or quality control problems in their manufacturing operations, or fail to meet our future requirements for timely delivery, our ability to ship certain of our hardware systems products to our customers could be impaired and our hardware systems business could be harmed.

We plan to simplify our supply chain processes by reducing the number of third party manufacturing partners on which Sun had historically relied and the number of locations where these third party manufacturers will build our hardware systems products. We will therefore become more dependent on a fewer number of these manufacturing partners and locations. If these partners experience production problems or delays or cannot meet our demand for products, we may not be able to find alternate manufacturing sources in a timely or cost effective manner, if at all. If we are required to change third party manufacturers, our ability to meet our scheduled hardware products deliveries to our customers could be adversely affected, which could cause the loss of sales and existing or potential customers, delayed revenue recognition or an increase in our hardware systems products expenses which could adversely affect the margins of our hardware systems business.

We may not be able to protect our intellectual property rights. We rely on copyright, trademark, patent and trade secret laws, confidentiality procedures, controls and contractual commitments to protect our intellectual property rights. Despite our efforts, these protections may be limited. Unauthorized third parties may try to copy or reverse engineer portions of our products or otherwise obtain and use our intellectual property. Any patents owned by us may be invalidated, circumvented or challenged. Any of our pending or future patent applications, whether or not being currently challenged, may not be issued with the scope of the claims we seek, if at all. In addition, the laws of some countries do not provide the same level of protection of our intellectual property rights as do the laws and courts of the United States. If we cannot protect our intellectual property rights against unauthorized copying or use, or other misappropriation, we may not remain competitive.

Third parties have claimed and, in the future, may claim infringement or misuse of intellectual property rights and/or breach of license agreement provisions. We periodically receive notices from, or have lawsuits filed against us by, others claiming infringement or other misuse of their intellectual property rights and/or breach of our agreements with them. We expect the number of such claims will increase as:

- we continue to acquire companies and expand into new businesses;
- the number of products and competitors in our industry segments grows;

- the functionality of products overlap;
- the use and support of third-party code (including open source code) becomes more prevalent in the industry; and
- the volume of issued patents continues to increase.

Responding to any such claim, regardless of its validity, could:

- be time consuming, costly and result in litigation;
- divert management's time and attention from developing our business;
- require us to pay monetary damages or enter into royalty and licensing agreements that we would not normally find acceptable;
- require us to stop selling or to redesign certain of our products;
- require us to release source code to third parties, possibly under open source license terms;
- require us to satisfy indemnification obligations to our customers; or
- otherwise adversely affect our business, results of operations, financial condition or cash flows.

Specific patent infringement cases are discussed under Note 18 of Notes to Consolidated Financial Statements.

We may lose key employees or may be unable to hire enough qualified employees. We rely on the continued service of our senior management, including our Chief Executive Officer and founder, members of our executive team and other key employees and the hiring of new qualified employees. In the technology industry, there is substantial and continuous competition for highly skilled business, product development, technical and other personnel. In addition, acquisitions could cause us to lose key personnel of the acquired companies or at Oracle. We may also experience increased compensation costs that are not offset by either improved productivity or higher prices. We may not be successful in recruiting new personnel and in retaining and motivating existing personnel. With rare exceptions, we do not have long-term employment or non-competition agreements with our employees. Members of our senior management team have left Oracle over the years for a variety of reasons, and we cannot assure you that there will not be additional departures, which may be disruptive to our operations.

We continually focus on improving our cost structure by hiring personnel in countries where advanced technical expertise is available at lower costs. When we make adjustments to our workforce, we may incur expenses associated with workforce reductions that delay the benefit of a more efficient workforce structure. We may also experience increased competition for employees in these countries as the trend toward globalization continues, which may affect our employee retention efforts and increase our expenses in an effort to offer a competitive compensation program. Our compensation program includes stock options, which are an important tool in attracting and retaining employees in our industry. If our stock price performs poorly, it may adversely affect our ability to retain or attract employees. In addition, because we expense all stock-based compensation, we may in the future change our stock-based and other compensation practices. Some of the changes we consider from time to time include a reduction in the number of employees granted options, a reduction in the number of options granted per employee and a change to alternative forms of stock-based compensation. Any changes in our compensation practices or changes made by competitors could affect our ability to retain and motivate existing personnel and recruit new personnel.

We may need to change our pricing models to compete successfully. The intense competition we face in the sales of our products and services and general economic and business conditions can put pressure on us to change our prices. If our competitors offer deep discounts on certain products or services or develop products that the marketplace considers more valuable, we may need to lower prices or offer other favorable terms in order to compete successfully. Any such changes may reduce margins and could adversely affect operating results. Our software license updates and product support fees and hardware systems support fees are generally priced as a percentage of our net new software license fees and net new hardware systems products fees, respectively. Our competitors may offer lower pricing on their support offerings, which could put pressure on us to further discount our new license prices.

Any broad-based change to our prices and pricing policies could cause our revenues to decline or be delayed as our sales force implements and our customers adjust to the new pricing policies. Some of our competitors may bundle products for promotional purposes or as a long-term pricing strategy or provide guarantees of prices and product implementations. These practices could, over time, significantly constrain the prices that we can charge for certain of our products. If we do not adapt our pricing models to reflect changes in customer use of our products or changes in customer demand, our revenues could decrease. Additionally, increased distribution of applications through application service providers, including SaaS providers, may reduce the average price for our products or adversely affect other sales of our products, reducing our revenues unless we can offset price reductions with volume increases. The increase in open source software distribution may also cause us to change our pricing models.

Charges to earnings resulting from acquisitions may adversely affect our operating results. Under business combination accounting standards pursuant to ASC 805, *Business Combinations*, we recognize the identifiable assets acquired, the liabilities assumed, and any non-controlling interests in acquired companies generally at their acquisition date fair values and, in each case, separately from goodwill. Goodwill as of the acquisition date is measured as the excess amount of consideration transferred, which is also generally measured at fair value, and the net of the acquisition date amounts of the identifiable assets acquired and the liabilities assumed. Our estimates of fair value are based upon assumptions believed to be reasonable but which are inherently uncertain. After we complete an acquisition, the following factors could result in material charges and adversely affect our operating results and may adversely affect our cash flows:

- costs incurred to combine the operations of companies we acquire, such as transitional employee expenses and employee retention, redeployment or relocation expenses;
- impairment of goodwill or intangible assets;
- amortization of intangible assets acquired;
- a reduction in the useful lives of intangible assets acquired;
- identification of or changes to assumed contingent liabilities, both income tax and non-income tax related, after our final determination of the amounts for these contingencies or the conclusion of the measurement period (generally up to one year from the acquisition date), whichever comes first;
- charges to our operating results to eliminate certain duplicative pre-merger activities, to restructure our operations or to reduce our cost structure;
- charges to our operating results resulting from expenses incurred to effect the acquisition; and
- charges to our operating results due to the expensing of certain stock awards assumed in an acquisition.

Substantially all of these costs will be accounted for as expenses that will decrease our net income and earnings per share for the periods in which those costs are incurred. Charges to our operating results in any given period could differ substantially from other periods based on the timing and size of our future acquisitions and the extent of integration activities. A more detailed discussion of our accounting for these and other items is presented in the "Critical Accounting Policies and Estimates" section of Management's Discussion and Analysis of Financial Condition and Results of Operations (Item 7).

Our periodic workforce restructurings can be disruptive. We have in the past restructured or made other adjustments to our workforce, including our direct sales force on which we rely heavily, in response to management changes, product changes, performance issues, acquisitions and other internal and external considerations. In the past, sales force and other restructurings have generally resulted in a temporary lack of focus and reduced productivity. These effects could recur in connection with future acquisitions and other restructurings and our revenues could be negatively affected.

We might experience significant errors or security flaws in our software products and services. Despite testing prior to their release, software products frequently contain errors or security flaws, especially when first introduced or when new versions are released. The detection and correction of any security flaws can be time consuming and costly. Errors in our software products could affect the ability of our products to work with other

hardware or software products, could delay the development or release of new products or new versions of products and could adversely affect market acceptance of our products. If we experience errors or delays in releasing new software products or new versions of software products, we could lose revenues. In addition, we run our own business operations, Oracle On Demand and other outsourcing services, support and consulting services, on our products and networks and any security flaws, if exploited, could affect our ability to conduct internal business operations. End users, who rely on our software products and services for applications that are critical to their businesses, may have a greater sensitivity to product errors and security vulnerabilities than customers for software products generally. Software product errors and security flaws in our products or services could expose us to product liability, performance and/or warranty claims as well as harm our reputation, which could impact our future sales of products and services. In addition, we may be legally required to publicly report security breaches of our services, which could adversely impact future business prospects for those services.

We may not receive significant revenues from our current research and development efforts for several years, if at all. Developing and localizing software and hardware products is expensive, and the investment in product development often involves a long return on investment cycle. We have made and expect to continue to make significant investments in research and development and related product opportunities. Accelerated product introductions and short product life cycles require high levels of expenditures for research and development that could adversely affect our operating results if not offset by revenue increases. We believe that we must continue to dedicate a significant amount of resources to our research and development efforts to maintain our competitive position. However, we do not expect to receive significant revenues from these investments for several years, if at all.

Our sales to government clients subject us to risks including early termination, audits, investigations, sanctions and penalties. We derive revenues from contracts with the U.S. government, state and local governments and their respective agencies, which may terminate most of these contracts at any time, without cause. There is increased pressure for governments and their agencies, both domestically and internationally, to reduce spending. Our federal government contracts are subject to the approval of appropriations being made by the U.S. Congress to fund the expenditures under these contracts. Similarly, our contracts at the state and local levels are subject to government funding authorizations. Additionally, government contracts are generally subject to audits and investigations which could result in various civil and criminal penalties and administrative sanctions, including termination of contracts, refund of a portion of fees received, forfeiture of profits, suspension of payments, fines and suspensions or debarment from future government business.

We may face numerous risks in connection with our strategic alliance with Fujitsu. Prior to our acquisition of Sun, Sun had entered into a number of agreements with Fujitsu Limited with respect to collaborative sales and marketing efforts and the joint development and manufacturing of Sun's server products known as the Advanced Product Line and branded as the Sun SPARC Enterprise line of servers. The agreements contemplate that Sun and Fujitsu dedicate substantial financial and human resources to this relationship. As a result, the future performance of our hardware systems business may be impacted by the success or failure of this relationship.

If we, as successors to Sun, fail to satisfy certain development or supply obligations under the agreements, or if we otherwise violate the terms of the agreements, we may be subject to contractual or legal penalties. Further, if Fujitsu encounters potential problems in its business, such as intellectual property infringement claims, supply difficulties, or difficulties in meeting development milestones or financial challenges, these problems could impact our strategic relationship with Fujitsu and could result in a material adverse effect on our hardware systems business. There can be no assurance that the strategic relationship with Fujitsu will be successful, or that the economic terms of the agreements establishing the relationship will ultimately prove to be favorable to us. The occurrence of any of these risks could have a material adverse effect on our hardware systems business.

Business disruptions could affect our operating results. A significant portion of our research and development activities and certain other critical business operations are concentrated in a few geographic areas. We are a highly automated business and a disruption or failure of our systems could cause delays in completing sales and providing services, including some of our On Demand offerings. A major earthquake, fire or other catastrophic event that results in the destruction or disruption of any of our critical business or information technology systems could severely affect our ability to conduct normal business operations and, as a result, our future operating results could be materially and adversely affected.

There are risks associated with our outstanding indebtedness. As of May 31, 2010, we had an aggregate of $14.7 billion of outstanding indebtedness that will mature between the remainder of calendar 2010 and calendar 2039, and we may incur additional indebtedness in the future. Our ability to pay interest and repay the principal for our indebtedness is dependent upon our ability to manage our business operations, generate sufficient cash flows to service such debt and the other factors discussed in this section. There can be no assurance that we will be able to manage any of these risks successfully.

We may also need to refinance a portion of our outstanding debt as it matures. There is a risk that we may not be able to refinance existing debt or that the terms of any refinancing may not be as favorable as the terms of our existing debt. Furthermore, if prevailing interest rates or other factors at the time of refinancing result in higher interest rates upon refinancing, then the interest expense relating to that refinanced indebtedness would increase. Should we incur future increases in interest expense, our ability to utilize certain of our foreign tax credits to reduce our U.S. federal income tax could be limited, which could unfavorably affect our provision for income taxes and effective tax rate. In addition, changes by any rating agency to our outlook or credit rating could negatively affect the value of both our debt and equity securities and increase the interest amounts we pay on outstanding or future debt. These risks could adversely affect our financial condition and results of operations.

Adverse litigation results could affect our business. We are subject to various legal proceedings. Litigation can be lengthy, expensive and disruptive to our operations, and results cannot be predicted with certainty. An adverse decision could result in monetary damages or injunctive relief that could affect our business, operating results or financial condition. Additional information regarding certain of the lawsuits we are involved in is discussed under Note 18 of Notes to Consolidated Financial Statements.

We may have exposure to additional tax liabilities. As a multinational corporation, we are subject to income taxes as well as non-income based taxes, in both the United States and various foreign jurisdictions. Significant judgment is required in determining our worldwide provision for income taxes and other tax liabilities.

Changes in tax laws or tax rulings may have a significantly adverse impact on our effective tax rate. For example, proposals for fundamental U.S. international tax reform, such as certain proposals by President Obama's Administration, if enacted, could have a significant adverse impact on our effective tax rate.

In the ordinary course of a global business, there are many intercompany transactions and calculations where the ultimate tax determination is uncertain. We are regularly under audit by tax authorities. Our intercompany transfer pricing is currently being reviewed by the IRS and by foreign tax jurisdictions and will likely be subject to additional audits in the future. We previously negotiated three successive unilateral Advance Pricing Agreements with the IRS that cover many of our intercompany transfer pricing issues and preclude the IRS from making a transfer pricing adjustment within the scope of the agreements. These agreements are effective for fiscal years through May 31, 2006. We have submitted to the IRS a request for another renewal of this Advance Pricing Agreement for the years ending May 31, 2007 through May 31, 2011. However, these agreements do not cover all elements of our transfer pricing and do not bind tax authorities outside the United States. We have finalized two bilateral Advance Pricing Agreements, one of which was effective for the years ending May 31, 2002 through May 31, 2006 and we have submitted a request for a renewal of this agreement for the years ending May 31, 2007 through May 31, 2011. There can be no guarantee that such negotiations will result in an agreement. The additional bilateral agreement covers the period from June 1, 2001 through January 25, 2008.

Although we believe that our tax estimates are reasonable, there is no assurance that the final determination of tax audits or tax disputes will not be different from what is reflected in our historical income tax provisions and accruals.

We are also subject to non-income based taxes, such as payroll, sales, use, value-added, net worth, property and goods and services taxes, in both the United States and various foreign jurisdictions. We are regularly under audit by tax authorities with respect to these non-income based taxes and may have exposure to additional non-income based tax liabilities. Our acquisition activities have increased our non-income based tax exposures, particularly with our entry into a new hardware systems business resulting from our acquisition of Sun, which increased the volume and complexity of laws and regulations that we are subject to and with which we must comply.

Oracle On Demand and CRM On Demand may not be successful. We offer Oracle On Demand outsourcing services for our applications and database technology, delivered at our data center facilities, select partner data

centers or customer facilities. We also offer CRM On Demand, which is a service offering that provides our customers with our CRM software functionality delivered via a hosted solution that we manage. These business models continue to evolve, and we may not be able to compete effectively, generate significant revenues, develop them into profitable businesses or maintain their profitability. We incur expenses associated with the infrastructures and marketing of our Oracle On Demand and CRM On Demand businesses in advance of our ability to recognize the revenues associated with these offerings. These businesses are subject to a variety of additional risks, including:

- we manage critical customer applications, data and other confidential information through Oracle On Demand and CRM On Demand; accordingly, we face increased exposure to significant damage claims and risk to Oracle's brand and future business prospects in the event of system failures, inadequate disaster recovery or loss or misappropriation of customer confidential information;

- we may face regulatory exposure in certain areas such as data privacy, data security and export compliance;

- the laws and regulations applicable to hosted service providers are unsettled, particularly in the areas of privacy and security and use of global resources; changes in these laws could affect our ability to provide services from or to some locations and could increase both the costs and risks associated with providing the services;

- demand for these services may not meet our expectations and may be affected by customer and media concerns about security risks, international transfers of data, government or other third-party access to data, and/or use of outsourced services providers more generally; and

- our offerings may require large fixed costs for data centers, computers, network infrastructure, security and otherwise, and we may not be able to generate sufficient revenues to offset these costs and generate acceptable operating margins from these offerings.

Environmental laws and regulations subject us to a number of risks and could result in significant liabilities and costs. Some of our operations acquired from Sun are subject to state, federal and international laws governing protection of the environment, proper handling and disposal of materials used to manufacture our products, human health and safety, and regulating the use of certain chemical substances. We endeavor to comply with these environmental laws, yet compliance with such laws could increase our product design, development, procurement and manufacturing costs, limit our ability to manage excess and obsolete non-compliant inventory, change our sales activities, or otherwise impact future financial results of our hardware systems business. Any violation of these laws can subject us to significant liability, including fines, penalties, and possible prohibition of sales of our products into one or more states or countries, and result in a material adverse effect on the financial condition or results of operation of our hardware systems business. A significant portion of Sun's hardware systems revenues historically has come from international sales. Recent environmental legislation within the European Union (EU), including the EU Directive on Restriction of Hazardous Substances (RoHS) and Waste Electrical and Electronic Equipment Directive (WEEE Directive), as well as China's regulation on Management Methods for Controlling Pollution Caused by Electronic Information Products may increase our cost of doing business internationally and impact our hardware systems revenues from EU countries and China as we endeavor to comply with and implement these requirements. In addition, similar environmental legislation has been or may be enacted in other jurisdictions, the cumulative impact of which could be significant.

Our stock price could become more volatile and your investment could lose value. All of the factors discussed in this section could affect our stock price. The timing of announcements in the public market regarding new products, product enhancements or technological advances by our competitors or us, and any announcements by us of acquisitions, major transactions, or management changes could also affect our stock price. Changes in the amounts and frequency of share repurchases or dividends could adversely affect our stock price. Our stock price is subject to speculation in the press and the analyst community, changes in recommendations or earnings estimates by financial analysts, changes in investors' or analysts' valuation measures for our stock, our credit ratings and market trends unrelated to our performance. A significant drop in our stock price could also expose us to the risk of securities class actions lawsuits, which could result in substantial costs and divert management's attention and resources, which could adversely affect our business.

Item 1B. Unresolved Staff Comments

None.

Item 2. Properties

Our properties consist of owned and leased office facilities for sales, support, research and development, consulting and administrative personnel. Our headquarters facility consists of approximately 2.0 million square feet in Redwood City, California. We also own or lease office facilities for current use consisting of approximately 26.0 million square feet in various other locations in the United States and abroad. Approximately 6.1 million square feet or 22% of total owned and leased space is sublet or is being actively marketed for sublease or disposition.

Item 3. Legal Proceedings

The material set forth in Note 18 of Notes to Consolidated Financial Statements in Item 15 of this Annual Report on Form 10-K is incorporated herein by reference.

Item 4. (Removed and Reserved)

Item 5. Market for Registrant's Common Equity, Related Stockholder Matters and Issuer Purchases of Equity Securities

Our common stock is traded on the NASDAQ Global Select Market under the symbol "ORCL" and has been traded on NASDAQ since our initial public offering in 1986. According to the records of our transfer agent, we had 18,059 stockholders of record as of May 31, 2010. The following table sets forth the low and high sale price of our common stock, based on the last daily sale, in each of our last eight fiscal quarters.

	Fiscal 2010		Fiscal 2009	
	Low Sale Price	High Sale Price	Low Sale Price	High Sale Price
Fourth Quarter	$ 21.91	$ 26.48	$ 13.85	$ 19.79
Third Quarter	$ 21.91	$ 25.34	$ 15.44	$ 18.41
Second Quarter	$ 20.34	$ 22.86	$ 15.40	$ 21.55
First Quarter	$ 19.69	$ 22.33	$ 20.25	$ 23.52

We declared and paid cash dividends totaling $0.20 per outstanding common share over the course of fiscal 2010. We declared and paid a cash dividend of $0.05 per outstanding common share for the first time in our history during our fourth quarter of fiscal 2009.

In June 2010, our Board of Directors declared a quarterly cash dividend of $0.05 per share of outstanding common stock payable on August 4, 2010 to stockholders of record as of the close of business on July 14, 2010. We currently expect to continue paying comparable cash dividends on a quarterly basis; however, future declarations of dividends and the establishment of future record and payment dates are subject to the final determination of our Board of Directors.

For equity compensation plan information, please refer to Item 12 in Part III of this Annual Report.

Stock Repurchase Programs

Our Board of Directors has approved a program for us to repurchase shares of our common stock. On October 20, 2008, we announced that our Board of Directors had approved the expansion of our repurchase program by $8.0 billion and as of May 31, 2010, approximately $5.3 billion was available for share repurchases pursuant to our stock repurchase program.

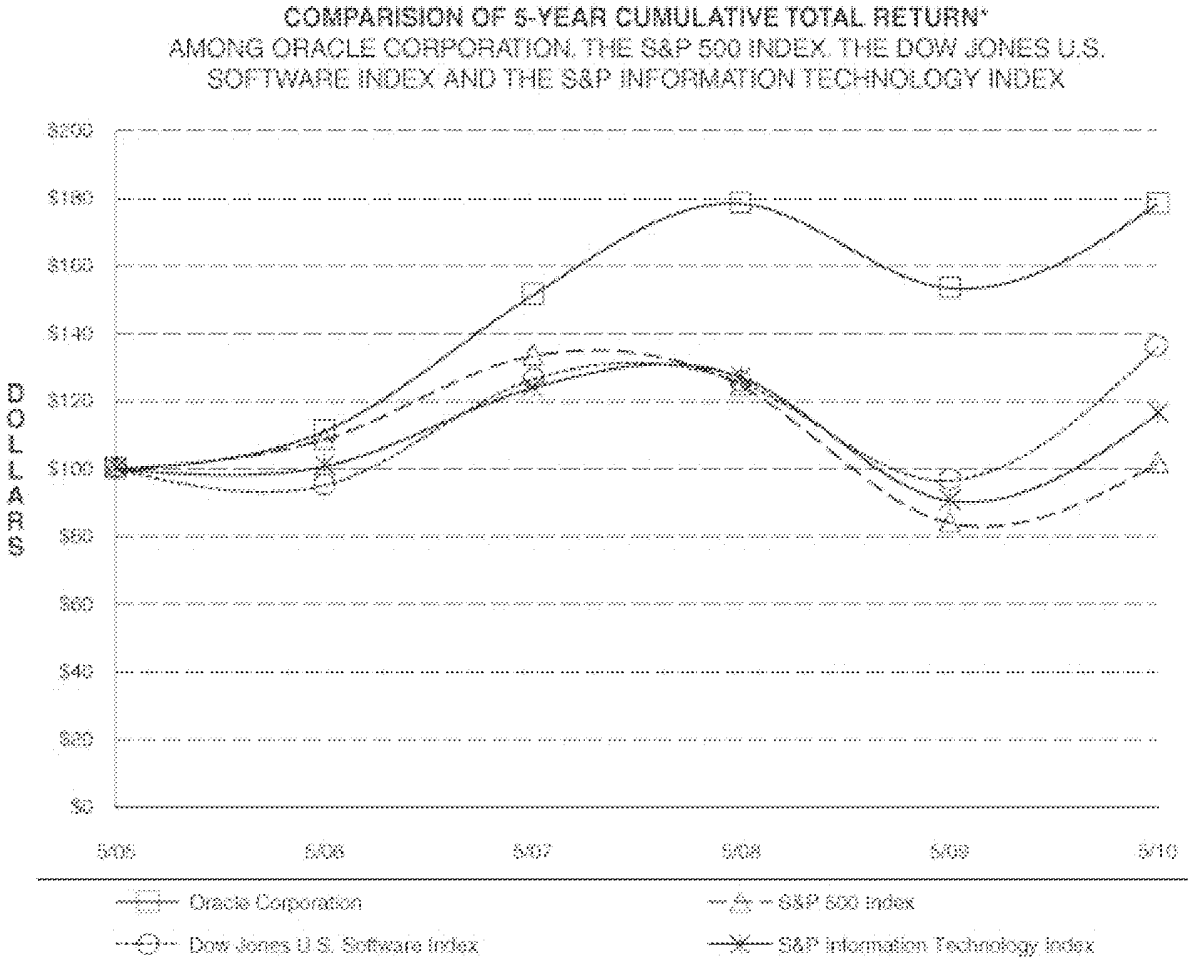
Our stock repurchase authorization does not have an expiration date and the pace of our repurchase activity will depend on factors such as our working capital needs, our cash requirements for acquisitions and dividend payments, our debt repayment obligations or repurchases of our debt, our stock price, and economic and market conditions. Our stock repurchases may be effected from time to time through open market purchases or pursuant to a Rule 10b5-1 plan. Our stock repurchase program may be accelerated, suspended, delayed or discontinued at any time.

The following table summarizes the stock repurchase activity for the three months ended May 31, 2010 and the approximate dollar value of shares that may yet be purchased pursuant to our stock repurchase program:

(in millions, except per share amounts)	Total Number of Shares Purchased	Average Price Paid per Share	Total Number of Shares Purchased as Part of Publicly Announced Programs	Approximate Dollar Value of Shares that May Yet Be Purchased Under the Programs
March 1, 2010—March 31, 2010	3.6	$ 25.23	3.6	$ 5,419.3
April 1, 2010—April 30, 2010	3.1	$ 26.05	3.1	$ 5,337.3
May 1, 2010—May 31, 2010	3.3	$ 23.57	3.3	$ 5,259.2
Total	10.0	$ 24.94	10.0	

Stock Performance Graph and Cumulative Total Return

The graph below compares the cumulative total stockholder return on our common stock with the cumulative total return of the S&P 500 Index, the Dow Jones U.S. Software Index and the S&P Information Technology Index for each of the last five fiscal years ended May 31, 2010, assuming an investment of $100 at the beginning of such period and the reinvestment of any dividends. As a result of our acquisition of Sun in fiscal 2010, we entered into a new hardware systems business and, for purposes of the stock performance graph below, we have changed one of the indexes to which we compare our cumulative total stockholder return from the Dow Jones U.S. Software Index to the S&P Information Technology Index. For the purposes of comparison, the stock performance graph below includes the new index and the previously reported index. The comparisons in the graphs below are based upon historical data and are not indicative of, nor intended to forecast, future performance of our common stock.



*$100 INVESTED ON MAY 31, 2005 IN STOCK OR
INDEX-INCLUDING REINVESTMENT OF DIVIDENDS.

	5/05	5/06	5/07	5/08	5/09	5/10
Oracle Corporation	100.00	111.09	151.41	178.44	153.45	178.33
S&P 500 Index	100.00	108.64	133.40	124.47	83.93	101.54
Dow Jones U.S. Software Index	100.00	95.10	126.43	125.50	96.52	136.22
S&P Information Technology Index	100.00	100.73	123.93	127.10	90.54	116.32

Item 6. Selected Financial Data

The following table sets forth selected financial data as of and for the last five fiscal years. This selected financial data should be read in conjunction with the consolidated financial statements and related notes included in Item 15 of this Annual Report. Over the last five fiscal years, we have acquired a number of companies including Sun Microsystems, Inc. in fiscal 2010, BEA Systems, Inc. in fiscal 2008, Hyperion Solutions Corporation in fiscal 2007 and Siebel Systems, Inc. in fiscal 2006. The results of our acquired companies have been included in our consolidated financial statements since their respective dates of acquisition and have contributed to our growth in revenues, income and earnings per share.

	As of and for the Year Ended May 31,				
(in millions, except per share amounts)	2010	2009	2008	2007	2006
Consolidated Statements of Operations Data:					
Total revenues	$ 26,820	$ 23,252	$ 22,430	$ 17,996	$ 14,380
Operating income	$ 9,062	$ 8,321	$ 7,844	$ 5,974	$ 4,736
Net income	$ 6,135	$ 5,593	$ 5,521	$ 4,274	$ 3,381
Earnings per share—basic	$ 1.22	$ 1.10	$ 1.08	$ 0.83	$ 0.65
Earnings per share—diluted	$ 1.21	$ 1.09	$ 1.06	$ 0.81	$ 0.64
Basic weighted average common shares outstanding	5,014	5,070	5,133	5,170	5,196
Diluted weighted average common shares outstanding	5,073	5,130	5,229	5,269	5,287
Cash dividends declared per common share	$ 0.20	$ 0.05	$ —	$ —	$ —
Consolidated Balance Sheets Data:					
Working capital[1]	$ 12,313	$ 9,432	$ 8,074	$ 3,496	$ 5,044
Total assets	$ 61,578[2]	$ 47,416	$ 47,268[2]	$ 34,572[2]	$ 29,029
Notes payable, current and other current borrowings[3]	$ 3,145	$ 1,001	$ 1,001	$ 1,358	$ 159
Notes payable and other non-current borrowings[4]	$ 11,510	$ 9,237	$ 10,235	$ 6,235	$ 5,735

(1) Total working capital sequentially increased in most periods primarily due to the favorable impact to our net current assets resulting from our net income generated during these periods and the issuance of $4.5 billion, $5.0 billion and $5.75 billion of long-term senior notes in fiscal 2010, fiscal 2008 and fiscal 2006, respectively. These increases were partially offset by cash used for acquisitions and repurchases of common stock in all periods presented and repayments of certain of our senior notes in fiscal 2010, 2009 and 2007 and dividend payments made in fiscal 2010 and 2009.

(2) Total assets increased as of May 31, 2010, 2008 and 2007 in comparison to the prior year period primarily due to goodwill and intangible assets arising from the acquisitions of Sun in fiscal 2010, BEA in fiscal 2008 and Hyperion in fiscal 2007, as well as our profitability in all periods presented. See Note 2 of Notes to Consolidated Financial Statements for additional information on our acquisitions.

(3) Notes payable, current and other current borrowings increased in fiscal 2010 primarily due to the reclassification of $2.2 billion of our senior notes due January 2011 as a current liability and our outstanding commercial paper notes of $881 million, partially offset by the repayment of $1.0 billion of senior notes that matured in May 2010. Notes payable, current and other current borrowings remained constant in fiscal 2009 due to repayment of $1.0 billion of senior notes that matured in May 2009 offset by the reclassification of $1.0 billion of senior notes that matured in fiscal 2010. Notes payable, current and other current borrowings decreased in fiscal 2008 due to repayments of amounts borrowed under our commercial paper program during fiscal 2007 partially offset by the reclassification of $1.0 billion of senior notes as a current liability that matured and were repaid in fiscal 2009.

(4) Notes payable and other non-current borrowings generally increased between fiscal 2006 and 2010 due to the issuances of $4.5 billion of long-term senior notes in fiscal 2010, $5.0 billion of long-term senior notes in fiscal 2008, and $2.0 billion of long-term senior notes in fiscal 2007, partially offset by redemptions of $1.5 billion in fiscal 2007. In fiscal 2010, 2009 and 2008, $2.2 billion, $1.0 billion and $1.0 billion, respectively, of non-current senior notes were reclassified as current liabilities. See Note 8 of Notes to Consolidated Financial Statements for additional information regarding our notes payable and other borrowings.

Item 7. Management's Discussion and Analysis of Financial Condition and Results of Operations

We begin Management's Discussion and Analysis of Financial Condition and Results of Operations with an overview of our key operating business segments and significant trends, including changes to our business as a result of our acquisition of Sun Microsystems, Inc. (Sun) in fiscal 2010. This overview is followed by a summary of our critical accounting policies and estimates that we believe are important to understanding the assumptions and judgments incorporated in our reported financial results. We then provide a more detailed analysis of our results of operations and financial condition.

Business Overview

We are the world's largest enterprise software company. As a result of our acquisition of Sun in January 2010, we are also a leading provider of hardware systems products and services. We develop, manufacture, market, distribute and service database and middleware software, applications software and hardware systems, consisting primarily of computer server and storage products, which are designed to help our customers manage and grow their business operations.

Our goal is to be the world's most complete, open and integrated enterprise software and hardware company. We offer customers scalable, reliable, secure and integrated software and hardware solutions that are designed to improve transactional efficiencies, adapt to an organization's unique needs and allow better ways to access and manage information and automate business processes at a lower total cost of ownership. We seek to be an industry leader in each of the specific product categories in which we compete and to expand into new and emerging markets.

We believe our internal, or organic, growth and continued innovation with respect to our software, hardware and services businesses are the foundation of our long-term strategic plan. In fiscal 2010, 2009 and 2008 we invested $3.3 billion, $2.8 billion and $2.7 billion, respectively, in research and development to enhance our existing portfolio of products and services and to develop new products, features and services. We also believe that an active acquisition program is an important element of our corporate strategy as it strengthens our competitive position, expands our customer base, provides greater scale to accelerate innovation, grows our revenues and earnings, and increases stockholder value. In recent years, we have invested billions of dollars to acquire a number of companies, products, services and technologies that add to, are complementary to or have otherwise enhanced our existing offerings. We expect to continue to acquire companies, products, services and technologies in furtherance of our corporate strategy.

We are organized into three businesses—software, hardware systems and services—which are further divided into seven operating segments. Each of these businesses and operating segments has unique characteristics and faces different opportunities and challenges. Although we report our actual results in U.S. Dollars, we conduct a significant number of transactions in currencies other than U.S. Dollars. Therefore, we present constant currency information to provide a framework for assessing how our underlying businesses performed excluding the effect of foreign currency rate fluctuations. An overview of our three businesses and seven operating segments follows.

Software Business

Our software business, which represented 77%, 81% and 80% of our total revenues in fiscal 2010, 2009 and 2008, respectively, is comprised of two operating segments: (1) new software licenses and (2) software license updates and product support. On a constant currency basis, we expect that our software business' total revenues generally will continue to increase due to continued demand for our software products and software license updates and product support offerings, including the high percentage of customers that renew their software license updates and product support contracts, and due to our acquisitions, which should allow us to grow our profits and continue to make investments in research and development.

New Software Licenses: We license our database and middleware as well as our applications software to businesses of many sizes, government agencies, educational institutions and resellers. As a result of our acquisition of Sun, we acquired certain software technologies, including Java, which is a global software development platform used in a wide range of computers, networks and devices.

The growth in new software license revenues that we report is affected by the strength of general economic and business conditions, governmental budgetary constraints, the competitive position of our software products, our acquisitions and foreign currency fluctuations. The substantial majority of our new software license business is also characterized by long sales cycles. The timing of a few large software license transactions can substantially affect our quarterly new software license revenues. Since our new software license revenues in a particular quarter can be difficult to predict as a result of the timing of a few large software license transactions, we believe that analysis of new software license revenues on a trailing 4-quarter period (as provided in our quarterly reports on Form 10-Q) provides additional visibility into the underlying performance of our new software license business. New software license revenues represented 28%, 31% and 34% of our total revenues in fiscal 2010, 2009 and 2008, respectively. The proportion of our new software license revenues relative to our total revenues in fiscal 2010 was affected by our entry into the hardware systems business as a result of our acquisition of Sun. Our new software license segment's margins have historically trended upward over the course of the four quarters within a particular fiscal year due to the historical upward trend of our new software license revenues over those quarterly periods and because the majority of our costs for this segment are predominantly fixed in the short term. However, our new software license segment's margins have been and will continue to be affected by the amortization of intangible assets associated with companies that we have acquired, including Sun.

Software License Updates and Product Support: Customers that purchase software license updates and product support are granted rights to unspecified product upgrades and maintenance releases issued during the support period, as well as technical support assistance. Substantially all of our customers renew their software license updates and product support contracts annually. The growth of software license updates and product support revenues is primarily influenced by three factors: (1) the percentage of our support contract customer base that renews its support contracts, (2) the amount of new support contracts sold in connection with the sale of new software licenses, and (3) the amount of support contracts assumed from companies we have acquired.

Software license updates and product support revenues, which represented 49%, 50% and 46% of our total revenues in fiscal 2010, 2009 and 2008, respectively, is our highest margin business unit. The proportion of our software license updates and product support revenues relative to our total revenues in fiscal 2010 was affected by our entry into the hardware systems business as a result of our acquisition of Sun. Support margins during fiscal 2010 were 85% and accounted for 80% of our total margins. Our software license update and product support margins have been affected by fair value adjustments relating to support obligations assumed in business combinations (described further below) and by amortization of intangible assets. However, over the longer term, we believe that software license updates and product support revenues and margins will grow for the following reasons:

- substantially all of our customers, including customers from acquired companies, renew their support contracts when eligible for renewal;
- substantially all of our customers purchase software license updates and product support contracts when they buy new software licenses, resulting in a further increase in our support contract base. Even if new software license revenues growth was flat, software license updates and product support revenues would continue to grow in comparison to the corresponding prior year periods assuming renewal and cancellation rates and foreign currency rates remained relatively constant since substantially all new software license transactions result in the sale of software license updates and product support contracts, which add to our support contract base; and
- our acquisitions have increased our support contract base, as well as the portfolio of products available to be licensed and supported.

We record adjustments to reduce support obligations assumed in business combinations to their estimated fair values at the acquisition dates. As a result, as required by business combination accounting rules, we did not recognize software license updates and product support revenues related to support contracts that would have been otherwise recorded by the acquired businesses as independent entities in the amount of $86 million, $243 million and $179 million in fiscal 2010, 2009 and 2008, respectively. To the extent underlying support contracts are renewed with us following an acquisition, we will recognize the revenues for the full value of the support contracts over the support periods, the majority of which are one year.

Hardware Systems Business

As a result of our acquisition of Sun in January 2010, we entered into a new hardware systems business. Our hardware systems business consists of two operating segments: (1) hardware systems products and (2) hardware systems support. Our hardware business represented 9% of our total revenues in fiscal 2010 and we expect that it will continue to add a significant amount of revenues and expenses to our results of operations in comparison to our historical operating results. We expect our hardware business to have lower operating margins as a percentage of revenues than our software business due to the incremental costs we incur to produce and distribute these products and to provide support services, including direct materials and labor costs. We expect to make investments in research and development to improve existing hardware products and services or develop new hardware products and services.

To produce our hardware products, we rely on both our internal manufacturing operations as well as third party manufacturing partners. Our internal manufacturing operations consist primarily of final assembly, test and quality control of enterprise and data center servers and storage systems. For all other manufacturing, we rely on third party manufacturing partners. We distribute most of our hardware products either from our facilities or partner facilities. One of our main goals is to reduce costs by simplifying our manufacturing processes through increased standardization of components across product types, through a reduction of the number of assembly and distribution centers we rely on and through our transition to a "build-to-order" process in which products are built only after customers have placed firm orders. In addition, we are focusing on identifying hardware systems support processes that are intended to proactively identify and solve quality issues and to increase the amount of new hardware systems support contracts sold in connection with the sales of new hardware products.

Hardware Systems Products: Our hardware systems products consist primarily of computer server and storage product offerings and hardware-related software, including our Solaris operating system.

We offer a wide range of server systems using our SPARC microprocessor. Our SPARC servers are differentiated by their size, cost, form factor or configuration (rack, blade or stand-alone systems) and customer environments that they target (general purpose or specialized systems). Our mid- and large-size servers are designed to offer greater performance and lower total cost of ownership than mainframe systems for business critical applications and for customers having more computationally intensive needs. Our SPARC servers run the Solaris operating system and are designed for the most demanding mission critical enterprise environments at any scale.

We also offer a wide range of x86 servers differentiated by the same features as our SPARC servers. These x86 systems are substantially based upon microprocessor platforms from Intel Corporation and are also compatible with Solaris, Linux, Windows and other operating systems.

Our storage products are designed to securely manage, protect, archive and restore customers' data assets and consist of tape, disk and networking solutions for open systems and mainframe server environments.

Prior to our acquisition of Sun, the majority of Sun's hardware systems products were sold through indirect channels. Although we plan to continue to sell our hardware products through indirect channels, including independent distributors and value added resellers, we have begun enhancing direct sales coverage for our hardware systems products and intend that our direct sales force will sell proportionately more of our hardware systems products in the future than they do currently.

Our operating margins for our hardware systems products segment have been and will be affected by the amortization of intangible assets associated with our acquisition of Sun. In addition, business combination accounting rules require us to record acquired inventories at fair value, which generally will result in an unfavorable short-term impact to our expenses and operating margins as we sell these acquired inventories to customers in the post-combination period.

We have limited experience in predicting our quarterly hardware systems products revenues. The timing of customer orders and delays in our ability to timely manufacture or deliver a few large transactions could substantially affect the amount of hardware systems products revenues, expenses and operating margins that we report.

Hardware Systems Support: Customers that purchase our hardware systems products may also elect to purchase our hardware systems support offerings. Our hardware systems support offerings provide customers with software updates for the software components that are essential to the functionality of our systems and storage products, including Solaris, and can include product repairs, maintenance services, and technical support services. Typically, our hardware systems support contract arrangements are invoiced to the customer at the beginning of the support period and are one year in duration. The growth of our hardware systems support revenues is influenced by a number of factors, including the volume of purchases of hardware products, the mix of hardware products purchased, and the percentage of our hardware systems support contract customer base that renews its support contracts. All of these factors are heavily influenced by our customers' decisions to either maintain or upgrade their existing hardware systems' infrastructure to newly developed technologies that are available.

Our hardware systems support margins have been and will be affected by fair value adjustments relating to hardware systems support obligations assumed through, and by the amortization of intangible assets resulting from, our acquisition of Sun. As required by business combination accounting rules, we recorded adjustments to reduce our hardware systems support revenues for contracts assumed from our acquisition of Sun to their estimated fair values as of the acquisition date by an aggregate of $128 million for fiscal 2010. This amount would have been recorded as hardware systems support revenues by Sun as a standalone entity. To the extent underlying hardware systems support contracts are renewed with us following an acquisition, we will recognize the revenues for the full values of the hardware systems support contracts over the support periods.

Services Business

Our services business consists of consulting, On Demand and education. As a result of our acquisition of Sun, we expanded and enhanced our customer base and services offerings, which we believe will increase our revenues and expenses in comparison to recent periods. Our services business, which represented 14%, 19% and 20% of our total revenues in fiscal 2010, 2009 and 2008, respectively, has significantly lower margins than our software business. The proportion of our services revenues relative to our total revenues in fiscal 2010 was affected by our entry into the hardware systems business as a result of our acquisition of Sun.

Consulting: Our consulting line of business primarily provides services to customers in enterprise architecture design and implementation; business/IT strategy alignment; business process simplification; solution integration; and product implementation, enhancements, and upgrades. The amount of consulting revenues recognized tends to lag the amount of our software and hardware systems products revenues by several quarters since consulting services, if purchased, are typically segmentable from the products with which they relate and are performed after the customer's purchase of the products. Our consulting revenues are dependent upon general economic conditions and the level of product revenues, in particular the new software license sales of our application products. To the extent we are able to grow our products revenues, in particular our software application product revenues, we would also generally expect to be able to eventually grow our consulting revenues.

On Demand: On Demand includes our Oracle On Demand and our Advanced Customer Services offerings. As a result of our acquisition of Sun, we increased the volume and breadth of our On Demand services offerings, primarily with additional Advanced Customer Services offerings, including staff augmentation and managed services, to architect, implement and manage customer IT environments. We believe that our On Demand offerings provide our customers flexibility in how they manage their IT environments and an additional opportunity to lower their total cost of ownership and can therefore provide us with a competitive advantage. While we have grown the base of customers that purchase certain of our On Demand services through our organic growth and the acquisition of Sun, we continue to focus on managing our expenses to increase our margins and margins as a percentage of our revenues. We have made and plan to continue to make investments in our On Demand business to support current and future revenue growth, which historically has negatively impacted On Demand margins and could do so in the future.

Education: The purpose of our education services is to further the adoption and usage of our software and hardware products by our customers and to create opportunities to grow our product revenues. Education revenues are impacted by certain of our acquisitions (including our acquisition of Sun), general economic

conditions, personnel reductions in our customers' information technology departments, tighter controls over discretionary spending and greater use of outsourcing solutions. In recent periods, we believe the global economic environment has unfavorably affected customer demand for our education services, which has negatively impacted our revenues and margins.

Acquisitions

An active acquisition program is another important element of our corporate strategy. In recent years, we have invested billions of dollars to acquire a number of complementary companies, products, services and technologies including Sun in fiscal 2010 and BEA Systems, Inc. in fiscal 2008, among others. We believe our acquisition program supports our long-term strategic direction, strengthens our competitive position, expands our customer base, provides greater scale to accelerate innovation, grows our revenues and earnings, and increases stockholder value. We expect to continue to acquire companies, products, services and technologies in furtherance of our corporate strategy. See Note 2 of Notes to Consolidated Financial Statements included elsewhere in this Annual Report for additional information related to our recent acquisitions.

We believe we can fund our pending and future acquisitions with our internally available cash, cash equivalents and marketable securities, cash generated from operations, amounts available under our existing debt facilities, additional borrowings or from the issuance of additional securities. We estimate the financial impact of any potential acquisition with regard to earnings, operating margin, cash flow and return on invested capital targets before deciding to move forward with an acquisition.

Critical Accounting Policies and Estimates

Our consolidated financial statements are prepared in accordance with U.S. generally accepted accounting principles (GAAP) as set forth in the Financial Accounting Standards Board's Accounting Standards Codification (Codification) and consider the various staff accounting bulletins and other applicable guidance issued by the SEC. GAAP, as set forth within the Codification, requires us to make certain estimates, judgments and assumptions. We believe that the estimates, judgments and assumptions upon which we rely are reasonable based upon information available to us at the time that these estimates, judgments and assumptions are made. These estimates, judgments and assumptions can affect the reported amounts of assets and liabilities as of the date of the financial statements as well as the reported amounts of revenues and expenses during the periods presented. To the extent there are material differences between these estimates, judgments or assumptions and actual results, our financial statements will be affected. The accounting policies that reflect our more significant estimates, judgments and assumptions and which we believe are the most critical to aid in fully understanding and evaluating our reported financial results include the following:

- Revenue Recognition
- Business Combinations
- Goodwill and Intangible Assets—Impairment Assessments
- Accounting for Income Taxes
- Legal and Other Contingencies
- Stock-Based Compensation
- Allowances for Doubtful Accounts

In many cases, the accounting treatment of a particular transaction is specifically dictated by GAAP and does not require management's judgment in its application. There are also areas in which management's judgment in selecting among available alternatives would not produce a materially different result. Our senior management has reviewed the below critical accounting policies and related disclosures with the Finance and Audit Committee of the Board of Directors.

Revenue Recognition

Our sources of revenues include: (1) software, which includes new software license revenues and software license updates and product support revenues; (2) hardware systems, which includes the sale of hardware systems products including computer servers and storage products, and hardware systems support revenues; and (3) services, which include software and hardware related services including consulting, On Demand and education revenues.

Revenue Recognition for Software Products and Software Related Services (Software Elements)

New software license revenues represent fees earned from granting customers licenses to use our database, middleware and applications software, and exclude revenues derived from software license updates, which are included in software license updates and product support revenues. While the basis for software license revenue recognition is substantially governed by the accounting guidance contained in ASC 985-605, *Software-Revenue Recognition*, we exercise judgment and use estimates in connection with the determination of the amount of software and services revenues to be recognized in each accounting period.

For software license arrangements that do not require significant modification or customization of the underlying software, we recognize new software license revenues when: (1) we enter into a legally binding arrangement with a customer for the license of software; (2) we deliver the products; (3) customer payment is deemed fixed or determinable and free of contingencies or significant uncertainties; and (4) collection is probable. Substantially all of our new software license revenues are recognized in this manner.

Substantially all of our software license arrangements do not include acceptance provisions. However, if acceptance provisions exist as part of public policy, for example, in agreements with government entities where acceptance periods are required by law, or within previously executed terms and conditions that are referenced in the current agreement and are short-term in nature, we generally recognize revenues upon delivery provided the acceptance terms are perfunctory and all other revenue recognition criteria have been met. If acceptance provisions are not perfunctory (for example, acceptance provisions that are long-term in nature or are not included as standard terms of an arrangement), revenues are recognized upon the earlier of receipt of written customer acceptance or expiration of the acceptance period.

The vast majority of our software license arrangements include software license updates and product support contracts, which are entered into at the customer's option and are recognized ratably over the term of the arrangement, typically one year. Software license updates provide customers with rights to unspecified software product upgrades, maintenance releases and patches released during the term of the support period. Product support includes internet access to technical content, as well as internet and telephone access to technical support personnel. Software license updates and product support contracts are generally priced as a percentage of the net new software license fees. Substantially all of our customers renew their software license updates and product support contracts annually.

Revenue Recognition for Multiple-Element Arrangements—Software Products and Software Related Services (Software Arrangements)

We often enter into arrangements with customers that purchase both software related products and services from us at the same time, or within close proximity of one another (referred to as software related multiple-element arrangements). Such software related multiple-element arrangements include the sale of our software products, software license updates and product support contracts and other software related services whereby software license delivery is followed by the subsequent or contemporaneous delivery of the other elements. For those software related multiple-element arrangements, we have applied the residual method to determine the amount of license revenues to be recognized pursuant to ASC 985-605. Under the residual method, if fair value exists for undelivered elements in a multiple-element arrangement, such fair value of the undelivered elements is deferred with the remaining portion of the arrangement consideration recognized upon delivery of the software license or services arrangement. We allocate the fair value of each element of a software related multiple-element arrangement based upon its fair value as determined by our vendor specific objective evidence (VSOE—described further below), with any remaining amount allocated to the software license.

Revenue Recognition for Hardware Systems Products and Hardware Systems Related Services (Nonsoftware Elements)

Revenues from the sale of hardware systems products represent amounts earned primarily from the sale of computer servers and storage products. Our revenue recognition policy for these nonsoftware deliverables is based upon the accounting guidance contained in ASC 605, *Revenue Recognition*, and we exercise judgment and use estimates in connection with the determination of the amount of hardware systems products and hardware systems related services revenues to be recognized in each accounting period.

Revenues from the sales of hardware products are recognized when: (1) persuasive evidence of an arrangement exists; (2) we deliver the products and passage of the title to the buyer occurs; (3) the sale price is fixed or determinable; and (4) collection is reasonably assured. Revenues that are not recognized at the time of sale because the foregoing conditions are not met are recognized when those conditions are subsequently met. When applicable, we reduce revenues for estimated returns or certain other incentive programs where we have the ability to sufficiently estimate the effects of these items. Where an arrangement is subject to acceptance criteria and the acceptance provisions are not perfunctory (for example, acceptance provisions that are long-term in nature or are not included as standard terms of an arrangement), revenues are recognized upon the earlier of receipt of written customer acceptance or expiration of the acceptance period.

Our hardware systems support offerings generally provide customers with software updates for the software components that are essential to the functionality of our systems and storage products and can also include product repairs, maintenance services, and technical support services. Hardware systems support contracts are entered into at the customer's option and are recognized ratably over the contractual term of the arrangements.

Revenue Recognition for Multiple-Element Arrangements—Hardware Systems Products and Hardware Systems Related Services (Nonsoftware Arrangements)

In the third quarter of fiscal 2010, we early adopted the provisions of Accounting Standards Update No. 2009-13, *Revenue Recognition (Topic 605) Multiple-Deliverable Revenue Arrangements* (ASU 2009-13) and Accounting Standards Update 2009-14, *Software (Topic 985)—Certain Revenue Arrangements that Include Software Elements* (ASU 2009-14). ASU 2009-13 amended existing accounting guidance for revenue recognition for multiple-element arrangements. To the extent a deliverable within a multiple-element arrangement is not accounted for pursuant to other accounting standards, including ASC 985-605, *Software-Revenue Recognition*, ASU 2009-13 establishes a selling price hierarchy that allows for the use of an estimated selling price (ESP) to determine the allocation of arrangement consideration to a deliverable in a multiple element arrangement where neither VSOE nor third-party evidence (TPE) is available for that deliverable. ASU 2009-14 modifies the scope of ASC 985-605 to exclude tangible products containing software components and nonsoftware components that function together to deliver the product's essential functionality. In addition, ASU 2009-14 provides guidance on how a vendor should allocate arrangement consideration to nonsoftware and software deliverables in an arrangement where the vendor sells tangible products containing software components that are essential in delivering the tangible product's functionality.

As a result of our early adoption of ASU 2009-13 and ASU 2009-14, we applied the provisions of these accounting standards updates as of the beginning of fiscal 2010. The impact of our adoption of ASU 2009-13 and ASU 2009-14 was not material to our results of operations for fiscal 2010.

We enter into arrangements with customers that purchase both nonsoftware related products and services from us at the same time, or within close proximity of one another (referred to as nonsoftware multiple-element arrangements). Each element within a nonsoftware multiple-element arrangement is accounted for as a separate unit of accounting provided the following criteria are met: the delivered products or services have value to the customer on a standalone basis; and for an arrangement that includes a general right of return relative to the delivered products or services, delivery or performance of the undelivered product or service is considered probable and is substantially controlled by us. We consider a deliverable to have standalone value if the product or service is sold separately by us or another vendor or could be resold by the customer. Further, our revenue arrangements generally do not include a general right of return relative to the delivered products. Where the aforementioned criteria for a separate unit of accounting are not met, the deliverable is combined with the

undelivered element(s) and treated as a single unit of accounting for the purposes of allocation of the arrangement consideration and revenue recognition. For those units of accounting that include more than one deliverable but are treated as a single unit of accounting, we generally recognize revenues over the delivery period. For the purposes of revenue classification of the elements that are accounted for as a single unit of accounting, we allocate revenue to hardware systems and services based on a rational and consistent methodology utilizing our best estimate of fair value of such elements.

For our nonsoftware multiple-element arrangements, we allocate revenue to each element based on a selling price hierarchy at the arrangement inception. The selling price for each element is based upon the following selling price hierarchy: VSOE if available, TPE if VSOE is not available, or ESP if neither VSOE nor TPE is available (a description as to how we determine VSOE, TPE and ESP is provided below). If a tangible hardware systems product includes software, we determine whether the tangible hardware systems product and the software work together to deliver the product's essential functionality and, if so, the entire product is treated as a nonsoftware deliverable. The total arrangement consideration is allocated to each separate unit of accounting for each of the nonsoftware deliverables using the relative selling prices of each unit based on the aforementioned selling price hierarchy. We limit the amount of revenue recognized for delivered elements to an amount that is not contingent upon future delivery of additional products or services or meeting of any specified performance conditions.

To determine the selling price in multiple-element arrangements, we establish VSOE of selling price using the price charged for a deliverable when sold separately and for software license updates and product support and hardware systems support, based on the renewal rates offered to customers. For nonsoftware multiple element arrangements, TPE is established by evaluating similar and interchangeable competitor products or services in standalone arrangements with similarly situated customers. If we are unable to determine the selling price because VSOE or TPE doesn't exist, we determine ESP for the purposes of allocating the arrangement by considering several external and internal factors including, but not limited to, pricing practices, margin objectives, competition, geographies in which we offer our products and services, internal costs and stage of the product lifecycle. The determination of ESP is made through consultation with and approval by our management, taking into consideration our go-to-market strategy. As our, or our competitors', pricing and go-to-market strategies evolve, we may modify our pricing practices in the future, which could result in changes to our determination of VSOE, TPE and ESP. As a result, our future revenue recognition for multiple-element arrangements could differ materially from our results in the current period. Selling prices are analyzed on an annual basis or more frequently if we experience significant changes in our selling prices.

Revenue Recognition Policies Applicable to both Software and Nonsoftware Elements

Revenue Recognition for Multiple-Element Arrangements –Arrangements with Software and Nonsoftware Elements

We also enter into multiple-element arrangements that may include a combination of our various software related and nonsoftware related products and services offerings including hardware systems products, hardware systems support, new software licenses, software license updates and product support, consulting, On Demand and education. In such arrangements, we first allocate the total arrangement consideration based on the relative selling prices of the software group of elements as a whole and to the nonsoftware elements. We then further allocate consideration within the software group to the respective elements within that group following the guidance in ASC 985-605 and our policies described above. After the arrangement consideration has been allocated to the elements, we account for each respective element in the arrangement as described above.

Other Revenue Recognition Policies Applicable to Software and Nonsoftware Elements

Many of our software arrangements include consulting implementation services sold separately under consulting engagement contracts and are included as a part of our services business. Consulting revenues from these arrangements are generally accounted for separately from new software license revenues because the arrangements qualify as services transactions as defined in ASC 985-605. The more significant factors considered in determining whether the revenues should be accounted for separately include the nature of services (i.e. consideration of whether the services are essential to the functionality of the licensed product), degree of risk, availability of services from other vendors, timing of payments and impact of milestones or acceptance

criteria on the realizability of the software license fee. Revenues for consulting services are generally recognized as the services are performed. If there is a significant uncertainty about the project completion or receipt of payment for the consulting services, revenues are deferred until the uncertainty is sufficiently resolved. We estimate the proportional performance on contracts with fixed or "not to exceed" fees on a monthly basis utilizing hours incurred to date as a percentage of total estimated hours to complete the project. If we do not have a sufficient basis to measure progress towards completion, revenues are recognized when we receive final acceptance from the customer. When total cost estimates exceed revenues, we accrue for the estimated losses immediately using cost estimates that are based upon an average fully burdened daily rate applicable to the consulting organization delivering the services. The complexity of the estimation process and factors relating to the assumptions, risks and uncertainties inherent with the application of the proportional performance method of accounting affects the amounts of revenues and related expenses reported in our consolidated financial statements. A number of internal and external factors can affect our estimates, including labor rates, utilization and efficiency variances and specification and testing requirement changes.

On Demand is comprised of Oracle On Demand and Advanced Customer Services and is a part of our services business. Oracle On Demand services are offered as standalone arrangements or as a part of arrangements to customers buying new software licenses or hardware systems products and services. Our On Demand services provide multi-featured software and hardware management and maintenance services for our software and hardware systems products delivered at our data center facilities, select partner data centers or customer facilities. Advanced Customer Services provide customers with services to architect, implement and manage customer IT environments including software and hardware systems product management services, industry-specific solution support centers and remote and on-site expert services. Depending upon the nature of the arrangement, revenues from On Demand services are recognized as services are performed or ratably over the term of the service period, which is generally one year or less.

Education revenues are a part of our services business and include instructor-led, media-based and internet-based training in the use of our software and hardware products. Education revenues are recognized as the classes or other education offerings are delivered.

If an arrangement contains multiple elements and does not qualify for separate accounting for the product and service transactions, then new software license revenues and/or hardware systems products revenues, including the costs of hardware systems products, are generally recognized together with the services based on contract accounting using either the percentage-of-completion or completed-contract method. Contract accounting is applied to any bundled software, hardware systems and services arrangements: (1) that include milestones or customer specific acceptance criteria that may affect collection of the software license or hardware systems product fees; (2) where consulting services include significant modification or customization of the software or hardware systems product; (3) where significant consulting services are provided for in the software license contract or hardware systems product contract without additional charge or are substantially discounted; or (4) where the software license or hardware systems product payment is tied to the performance of consulting services. For the purposes of revenue classification of the elements that are accounted for as a single unit of accounting, we allocate revenues to software and nonsoftware elements based on a rational and consistent methodology utilizing our best estimate of fair value of such elements.

We also evaluate arrangements with governmental entities containing "fiscal funding" or "termination for convenience" provisions, when such provisions are required by law, to determine the probability of possible cancellation. We consider multiple factors, including the history with the customer in similar transactions, the "essential use" of the software or hardware systems products and the planning, budgeting and approval processes undertaken by the governmental entity. If we determine upon execution of these arrangements that the likelihood of cancellation is remote, we then recognize revenues once all of the criteria described above have been met. If such a determination cannot be made, revenues are recognized upon the earlier of cash receipt or approval of the applicable funding provision by the governmental entity.

We assess whether fees are fixed or determinable at the time of sale and recognize revenues if all other revenue recognition requirements are met. Our standard payment terms are net 30 days. However, payment terms may vary based on the country in which the agreement is executed. Payments that are due within six months are

generally deemed to be fixed or determinable based on our successful collection history on such arrangements, and thereby satisfy the required criteria for revenue recognition.

While most of our arrangements for sales within our software and hardware systems businesses include short-term payment terms, we have a standard practice of providing long-term financing to creditworthy customers through our financing division. Since fiscal 1989, when our financing division was formed, we have established a history of collection, without concessions, on these receivables with payment terms that generally extend up to five years from the contract date. Provided all other revenue recognition criteria have been met, we recognize new software license revenues and hardware systems products revenues for these arrangements upon delivery, net of any payment discounts from financing transactions. We have generally sold receivables financed through our financing division on a non-recourse basis to third party financing institutions and we classify the proceeds from these sales as cash flows from operating activities in our consolidated statements of cash flows. We account for the sales of these receivables as "true sales" as defined in ASC 860, *Transfers and Servicing*.

In addition, we sell hardware products to leasing companies that, in turn, lease these products to end-users. In transactions where the leasing companies have no recourse to us in the event of default by the end-user, we recognize revenue at point of shipment or point of delivery, depending on the shipping terms and if all the other revenue recognition criteria have been met. In arrangements where the leasing companies have more than insignificant recourse to us in the event of default by the end-user (defined as recourse leasing), we recognize both the product revenue and the related cost of the product as the payments are made to the leasing company by the end-user, generally ratably over the lease term.

Our customers include several of our suppliers and on rare occasion, we have purchased goods or services for our operations from these vendors at or about the same time that we have sold our products to these same companies (Concurrent Transactions). Software license agreements or sales of hardware systems that occur within a three-month time period from the date we have purchased goods or services from that same customer are reviewed for appropriate accounting treatment and disclosure. When we acquire goods or services from a customer, we negotiate the purchase separately from any sales transaction, at terms we consider to be at arm's length, and settle the purchase in cash. We recognize new software license revenues or hardware systems product revenues from Concurrent Transactions if all of our revenue recognition criteria are met and the goods and services acquired are necessary for our current operations.

Business Combinations

In fiscal 2010, we adopted ASC 805, *Business Combinations*, which revised the accounting guidance that we were required to apply for our acquisitions in comparison to prior fiscal years. The underlying principles are similar to the previous guidance and require that we recognize separately from goodwill the assets acquired and the liabilities assumed at their acquisition date fair values. Goodwill as of the acquisition date is measured as the excess of consideration transferred and the net of the acquisition date fair values of the assets acquired and the liabilities assumed. While we use our best estimates and assumptions as a part of the purchase price allocation process to accurately value assets acquired and liabilities assumed at the acquisition date, our estimates are inherently uncertain and subject to refinement. As a result, during the measurement period, which may be up to one year from the acquisition date, we record adjustments to the assets acquired and liabilities assumed, with the corresponding offset to goodwill. Upon the conclusion of the measurement period or final determination of the values of assets acquired or liabilities assumed, whichever comes first, any subsequent adjustments are recorded to our consolidated statements of operations.

As a result of adopting the revised accounting guidance provided for by ASC 805 as of the beginning of fiscal 2010, certain of our policies differ when accounting for acquisitions in fiscal 2010 and prospective periods in comparison to the accounting for acquisitions in fiscal 2009 and prior periods, including:

- the fair value of in-process research and development is recorded as an indefinite-lived intangible asset until the underlying project is completed, at which time the intangible asset is amortized over its estimated useful life, or abandoned, at which time the intangible asset is expensed (prior to fiscal 2010, in-process research and development was expensed as of the acquisition date);

- the direct transaction costs associated with a business combination are expensed as incurred (prior to fiscal 2010, direct transaction costs were included as part of the purchase price);

- the costs to exit or restructure certain activities of an acquired company are accounted for separately from the business combination (prior to fiscal 2010, restructuring and exist costs were included as a part of the assumed obligations in deriving the purchase price allocation); and

- any changes in estimates associated with income tax valuation allowances or uncertain tax positions after the measurement period are generally recognized as income tax expense with application of this policy also applied prospectively to all of our business combinations regardless of the acquisition date (prior to fiscal 2010, any such changes in estimates associated with income taxes were generally included as a part of the purchase price allocation indefinitely).

Accounting for business combinations requires our management to make significant estimates and assumptions, especially at the acquisition date with respect to intangible assets, support obligations assumed, estimated restructuring liabilities and pre-acquisition contingencies. Although we believe the assumptions and estimates we have made in the past have been reasonable and appropriate, they are based in part on historical experience and information obtained from the management of the acquired companies and are inherently uncertain.

Examples of critical estimates in valuing certain of the intangible assets we have acquired include but are not limited to:

- future expected cash flows from software license sales, hardware systems product sales, support agreements, consulting contracts, other customer contracts and acquired developed technologies and patents;

- expected costs to develop the in-process research and development into commercially viable products and estimated cash flows from the projects when completed;

- the acquired company's brand and competitive position, as well as assumptions about the period of time the acquired brand will continue to be used in the combined company's product portfolio; and

- discount rates.

Unanticipated events and circumstances may occur that may affect the accuracy or validity of such assumptions, estimates or actual results.

In connection with the purchase price allocations for our acquisitions, we estimate the fair value of both software license updates and product support and hardware systems support obligations assumed. The estimated fair values of these support obligations are determined utilizing a cost build-up approach. The cost build-up approach determines fair value by estimating the costs related to fulfilling the obligations plus a normal profit margin. The estimated costs to fulfill the support obligations are based on the historical direct costs related to providing the support services and to correct any errors in the products acquired. The sum of these costs and operating profit approximates, in theory, the amount that we would be required to pay a third party to assume the support obligation. We do not include any costs associated with selling efforts or research and development or the related fulfillment margins on these costs. Profit associated with any selling efforts is excluded because the acquired entities would have concluded those selling efforts on the support contracts prior to the acquisition date. We also do not include the estimated research and development costs in our fair value determinations, as these costs are not deemed to represent a legal obligation at the time of acquisition.

As a result, we did not recognize software license updates and product support revenues related to support contracts in the amounts of $86 million, $243 million and $179 million that would have been otherwise recorded by the acquired businesses as independent entities in fiscal 2010, 2009 and 2008, respectively. In addition, we did not recognize hardware systems support revenues related to hardware systems support contracts that would have otherwise been recorded by Sun as an independent entity in the amount of $128 million for fiscal 2010. Historically, substantially all of our customers, including customers from acquired companies, renew their software license updates and product support contracts when the contracts are eligible for renewal and we intend to focus our efforts on renewing acquired hardware systems support contracts. To the extent software support or hardware systems support contracts are renewed, we will recognize the revenues for the full value of the support contracts over the support periods, the substantial majority of which are one year.