INSIGHT: Understanding Your Oracle/Java License--Not Everything is as it Seems

By Arthur S. Beeman, Joel T. Muchmore and Molly A. Jones

Famously touting the cross-platform capabilities of allowing programmers to "write once, run anywhere," Oracle's Java is one of the most -- if not the most -- prevalent programming languages used today. However, accompanying this ease and popularity, Java-based programming may involve murky and unexpected licensing concerns. Bearing in mind Oracle's notoriety for aggressive licensing and auditing tactics, attentive general counsel should be aware and wary of its company's use of Java. In light of the recent reports that Oracle is adding Java to its expanding audit checklist, curtailing and accounting for your company's use of Java could save considerable time and expense when the inevitable Oracle audit commences.

Simply put, the popular perception, especially among computer-savvy IT personnel, is that Java is open source – that is, free for all to use, copy, and manipulate. It is not. And, aside from some very limited use, it has not been for some time. Stemming from both Java's history as an open-source platform, as well as from inherent ambiguities baked into Oracle's Java license, this misperception is understandable. Unfortunately, this prevalent misunderstanding can have serious and unexpected consequences on an unprepared company that becomes the target of one of Oracle's notoriously intrusive and disruptive licensing audits. And, according to some reports, Oracle has not only begun targeting Java licensees, it has hired a team of near two dozen employees that are specifically charged with enforcing Java licensing terms.

As we have noted before, preparation and attentiveness are the keys to surviving an Oracle audit. Though there is no substitute for the retention and advice of competent counsel, the following is intended to provide preliminary insights as to the misconceptions surrounding Java licensing, the ambiguities that could result in allegations of exceeding license scope, and a few tips that may mitigate the impact of a Java license audit.

Java is no longer a strictly free, open-source platform. The current misperception that Java is always free is rooted in its licensing history. Originally, Sun Microsystems offered Java under a proprietary license, but, beginning in 2007, Sun licensed Java for free under the GNU General Public License ("GNU GPL"). The GNU GPL enabled end users to modify, use, and copy Java software regardless of the application and without payment of royalties. Further, pursuant to the GNU GPL, distribution of software applications derived from Java was subject to the same free license terms.

For several years, Java was widely distributed on these terms. However, in 2010, Oracle acquired Sun and began offering Java Standard Edition (SE) under its Oracle Binary Code License Agreement ("BCL"), which provides that developers may use only *certain* features of Java SE for free, and even those features are free only if used for certain limited purposes. However, due to ambiguities in the BCL, even a careful user may have difficulty complying with the license terms. Specifically, per the terms of the BCL, Java SE is free only for use in "general-purpose computing" that does not entail use of the "commercial features." Failure to appreciate these nuances of "general purpose computing" and/or care to avoid use of "commercial features" can result in significant allegations of under-licensing.

''General-purpose computing'' is ambiguous and could lead to unforeseen alleged license fees. The BCL defines ''general purpose'' as use ''for general computing functions under end-user control (including, but not limited to, email, general-purpose internet browsing and office suite productivity tools).'' General-purpose computing is distinct from what Oracle refers to in the BCL as ''specialized embedded computers used in intelligent systems,'' which require the payment of license fees.

These terms are remarkably vague. And the problem with such vague and ambiguous definitions is compounded by the realities of modern-day computing, where the limits of ''general purpose,'' ''mobile,'' ''embedded,'' *etc*. are not easily determined. Aside from patent claim construction decisions defining these terms within the context of specific inventions, courts have not otherwise defined these terms, and the ambiguity of Oracle's Java license has not generally been tested. (In the business enterprise licensing context, Oracle's aggressive auditing tactics have been challenged in at least one suit, which was quickly dismissed before Oracle's license interpretation could face judicial scrutiny.)

With little guidance from the courts, this ambiguity provides Java licensees little protection against Oracle claiming that virtually all deployments of Java are ''specialized'' and, therefore, subject to fee-based licensing. To complicate matters more, because Oracle has not previously made enforcement of Java licenses a priority, the community of Java developers has not yet developed a basic outline of what use is fair game for free licensing and, conversely, what Oracle may consider fee-based use.

Oracle bundles Java's ''commercial features'' with free features in its standard Java SE download. Companies using Java SE can also inadvertently fall out of compliance with the BCL for failure to pay license fees on intentional or inadvertent use of Java's ''commercial features,'' regardless of whether such use is performed for ''general-purpose computing'' or not. The BCL expressly provides:

''THE LICENSE SET FORTH IN THIS SECTION 2 DOES NOT EXTEND TO THE COMMERCIAL FEATURES . . . . COMMERCIAL FEATURES. You may not use the Commercial Features for running Programs, Java applets or applications in your internal business operations or for any commercial or production purpose, or for any purpose other than as set forth in Sections B, C, D and E of these Supplemental Terms. If You want to use the Commercial Features for any purpose other than as permitted in this Agreement, You must obtain a separate license from Oracle.''

''Commercial features'' are identified in the Java SE Licensing Information User Manual under the ''Description of Product Editions and Permitted Features'' section, which currently includes critical features such as the Java Flight Recorder and Java Mission Control, JRockit Flight Recorder, and JRockit Mission Control. Notably, the commercial features are not enumerated in the BCL itself, raising the question of whether Oracle can unilaterally modify the list of paid features and sow further confusion amongst its licensees.

Importantly, because Oracle bundles different Java-related products, users cannot individually download the *free* features without also downloading certain *commercial* features. Moreover, there is no easy way to strip unlicensed commercial features from the Java Runtime Environment once they are installed. This inevitably has led to confusion in the software industry and, likely, the extensive and inadvertent unlicensed use of Java's commercial features.

License audits are key to Oracle's monetization of technologies like Java. It is no secret that Oracle aggressively audits its software licensees, including most significantly, enterprise database licensees. In fact, over the last decade Oracle has developed an industry-wide reputation for asserting what many believe to be untenable interpretations of its license agreements to support demands for alleged licensing shortfalls. Some have speculated that, in addition to yielding licensing revenue, Oracle audits have evolved into an opportunity to push sales of other Oracle offerings, such as its lagging cloud services, ostensibly to ''offset'' alleged unpaid license fees.

However, it is only in the last couple of years that Oracle has reportedly begun to ramp up its audits of Java developers. Perhaps ironically, it is the widespread popularity and adoption of Java under the GNU GPL (which allowed for the free flow of technological innovation that arguably supported the rise of big data, the cloud, and the internet of things) that has made Java users ripe and attractive targets for an aggressive auditing strategy.

Allegations of under-licensing Java can be no small matter. Despite the relatively low price tag for a Java license ($300 per named user plus license), Oracle's allegations of licensing shortfalls can metastasize rapidly. If an organization goes beyond so-called ''general purpose'' use of Java, Oracle assesses license fees on a ''Named User Plus'' and/or ''per processor'' basis, with additional costs for support that vary depending on the developer's use. Oracle also assesses license fees for distributed devices and/or software applications (''embedded'' Java) on a ''per processor'' basis. As with all Oracle software licenses, ''processor'' is defined as any processor in which the software is ''installed and/or running,'' with the actual count determined by the number of cores of each processor multiplied by the contractual ''core factor.'' At minimum, a company can be vulnerable to allegations of under-licensing for actual use as well as when embedded on distributed devices and software applications, which can add up quickly.

However, because Java is licensed on a ''per processor'' basis, use of Java is also potentially vulnerable to Oracle's oft-reported (and highly dubious) allegations regarding licensing in virtualized environments. In sum, by defining ''installed'' as ''available for use,'' Oracle has long argued that the capacity for live migration (the process of moving a running virtual machine or application between different physical machines) means that the programs are ''installed'' (a/k/a ''available for use'') on all processors across a company's virtual environments. (Please read *Mars v. Oracle*, authored by House of Brick Technologies, for an overview of Oracle's ''installed and/or running''/virtualization argument routinely employed during licensing audits.)

In the context of database licensing, Oracle has been rumored to advance this argument in order to allege licensing shortfalls of orders of magnitude with past due fees and support into the tens of millions of dollars (albeit based on a measurably more expensive product). Using the past as a guide, in-house counsel should be

wary that Oracle may use the same virtualization arguments in order to similarly compound allegations of Java under-licensing.

Oracle to discontinue public updates to Java SE 8. Oracle has recently announced that in January 2019 it will cease releasing public updates to the popular Java SE 8 for commercial (licensable). As a result, users must purchase Oracle support contracts to continue to receive bug fixes and security updates. Oracle will, however, continue to provide free public updates of Java SE 8 for non-commercial (non-licensable) use. Commercial users who wish to continue to receive support for Java SE 8 can opt to purchase one of various tiers of support: Premier, Extended, Sustaining, and Lifetime. Premier and Extended Support options only provide support up to certain predetermined dates according to Oracle's Java SE Support Roadmap.

Developers should consider these support pricing developments in determining whether to license currently-used versions of Java, upgrade to newer releases of Java in order to ensure the longest available support for Java-embedded products, or purchase higher-tier support licenses. Companies considering using newer releases of Java should contemplate leveraging such costly migrations to reduce ''true up'' payments for alleged past unlicensed use of Java.

Software developers should prepare early for Oracle's audit of Java licenses.

Because the potential to run afoul of Oracle's licenses for Java is extremely high, it is essential to understand the exposure to Java license fees that have accrued since Oracle acquired Java from Sun in 2010 and get out ahead of an Oracle licensing audit. There are three key components to mitigating the impact of an Oracle audit of Java licenses:

1. Conduct a self-audit. Review your Java assets to understand what versions of Java SE have been downloaded, what features have been used and are currently in use and on what devices, and the extent to which Java has been redistributed in shipped products. If it is necessary to uninstall prior versions of Java or any particular feature thereof, clearly document what was uninstalled and when.

2. Clearly articulate to developers and operations staff what Java SE features are and are not licensed. Ensure that the distinction between commercial features and free features is well-understood across the organization. Oracle will contend that accidental or non-use of commercial features is not a defense against the payment of license fees. It is therefore essential to minimize the inadvertent enabling of unlicensed commercial features.

3. Obtain legal counsel. Experienced licensing counsel may be necessary in order to analyze whether a company's use of Java is for general-purpose computing or otherwise. To the extent that Java may be considered outside of general-purpose computing, and to avoid the imposition of past and future license fees, obtain counsel to negotiate an amendment to the BCL to ensure current and future uses fall within the general-purpose computing exception to license fees. Leverage future acquisitions of Oracle technology to obtain Oracle's agreement that these additional uses constitute general-purpose computing.

* * *

At this time, where Oracle intends to take Java licensing and auditing is uncertain. However, after obtaining two major appellate victories in the *Oracle v. Google* matter – the first overturning a lower court finding that Java was not copyrightable and the second overturning a lower court finding that Google's use of Java in the Android operating system was fair use – it seems likely that Oracle feels emboldened with regard to the enforcement of Java licenses.

In this instance, an emboldened Oracle is cause for concern for all companies using Java. Because use of Java will vary depending on the application and development situation as well as the specifics of the Java licenses governing obligations to Oracle, experienced attorneys should be retained to ensure compliance with licenses, prevent overly intrusive audits by Oracle, and if necessary, renegotiate licenses to reflect true business needs.

Based in San Francisco, Arthur Beeman is a partner in Crowell & Moring's Litigation and Intellectual Property groups where he focuses on the trial and management of complex intellectual property and commercial litigation disputes, and has extensive experience handling matters in the telecommunications, internet, software, hardware, medical device, and construction machinery industries. Joel Muchmore is also a partner in the firm's Litigation and Intellectual Property groups. He has substantial experience with appellate issues, as a trial attorney, and in litigating patent infringement, licensing disputes, trademark, trade secret, copyright, unfair competition, and other complex commercial disputes. Molly A. Jones is an Intellectual Property and Litigation associate in Crowell & Moring's San Francisco office.