Five Steps to Android Readiness

Android State of the Union

Ready or not, Android is coming to the enterprise. The rise of Google's Android operating system has ushered a new wave of mobile devices and applications to market. Google reported on Feb 27, 2012 that:

- More than 850,000 new Android devices are activated each day.
- More than 300 million Android devices have been activated so far.
- More than 450,000 Android apps are now available on Google Play (previously Android Market).

Consumers typically buy Android smartphones and tablets for personal use, but as they become more dependent on mobile devices to run their lives, they also want to bring them to work.

IT professionals who only recently adopted iPhone and iPad may wonder: is Android worth it? While Android opens the door to further innovation and productivity, it presents new fragmentation and complexity challenges for IT.

Introducing Android is a balancing act between what users want and what IT requires. Users want choice in the devices they use at work and the ability to use the native experience of those devices; IT needs to secure the data stored on those devices and enforce company policies.

While Android opens the door to further innovation and productivity, it presents new fragmentation and complexity challenges for Mobile IT.

Mobile IT leaders with successful iOS deployments have learned that user experience is paramount, and that they must preserve the native experience without compromising security. This same lesson applies to Android, but the implementation challenge is greater.

This paper is the first in a series describing best practices for successful Android deployments. More than half of MobileIron's customers are using Android today, and their experiences form the basis for these best practices. This paper outlines the requirements to make Android enterprise-ready. Then, it details five steps a Mobile IT team should follow to make their enterprise, in turn, Android-ready.

Fragmentation: The Core Challenge

For IT, fragmentation means complexity and uncertainty. In 2011, this fragmentation substantially slowed the adoption of Android in the enterprise even as consumer adoption accelerated.

Because of its licensing model, Android appeals to a wide range of manufacturers, developers, and OEMs. HTC, LG, Motorola, Samsung, and others make hundreds of different Android products today—ranging from popular smartphones and tablets to Internet TV and industry-specific devices. Amazon Kindle is an example of a specialized Android device. To stand apart from competitors, manufacturers modify the user interface, hardware, display, memory, and processor of the devices they offer. The resulting landscape is fragmented, and each permutation creates a different user experience.

An example of fragmentation is the evolution of encryption for Android. In early 2011, Android 3.0 introduced encryption, but only for tablets. As a result, Samsung and Motorola provided their own Android encryption layers to satisfy the needs of the enterprise. In late 2011, Android 4.0 was announced, which brought encryption to any Android device with 4.0, but the upgrade path for many devices was not defined. So while encryption does in fact exist for Android, most IT professionals do not know for which operating system versions or device variants.

The Android apps arena is equally diverse. Unlike iOS, which has a single source of curated applications in Apple's App Store, Android supports multiple stores, including Google Play, Amazon Appstore, and many third-party marketplaces from wireless service providers and device manufacturers.

> These Android variances are surmountable but require proactive education and planning.

These variances across operating system, manufacturer, device, and application market are surmountable but require a greater level of proactive education and planning than iOS or BlackBerry.

Making Android Enterprise-Ready

Let's review the baseline capabilities that any mobile operating system must support before it can be broadly used for the enterprise:

a) **Lost device security**: Remote lock, wipe, and password policy enforcement for the device
b) **Encryption**: Monitoring and enforcement of data-at-rest encryption
c) **Enterprise e-mail**: Push e-mail client that supports ActiveSync and remote configuration
d) **Secure connectivity**: Remote configuration and management of Wi-Fi and VPN for data-in-motion
e) **Certificate storage**: Ability to store a certificate on the device and allow authorized applications to use it for identity, especially e-mail, Wi-Fi, and VPN
f) **Direct app installation**: Ability to install an internal company app directly on a device instead of having to post it first on a commercial marketplace such as Google Play

These basic operating system functions are the raw materials used by Mobile IT platforms such as MobileIron to build out the advanced management, security, and application capabilities an enterprise needs. They provide the minimum baseline.

Today, each Android device manufacturer is evolving independently along this baseline. Google is also adding these building blocks to the core Android operating system itself, though the timing of this appears to be further out.

As a result, Mobile IT should prepare for Android as multiple operating systems, not one, even though many capabilities are consistent.

Mobile IT should prepare for Android as multiple operating systems, not one, even though many capabilities are consistent.

Over the course of 2012, new virtualization technologies will also start playing a part in the Android ecosystem. These technologies promise the ability to have an enterprise-class Android experience running parallel to a consumer Android experience on the same device. But any virtual operating system running on Android will also need to support these six baseline capabilities to be considered for the enterprise, so the needs are the same even if the architecture is different.

Given all this uncertainty, some IT departments ask: **Do I really need to support Android in my company? The answer is yes**. In 2012, end-user pressure to adopt Android will increase dramatically due to both an explosion in the number and quality of Android devices available, and the rise of Bring Your Own Device (BYOD) initiatives that encourage employees to use their personal devices for work.

The rest of this paper provides a starting point to making your company or institution Android-ready.

Making the Enterprise Android-Ready

Step 1: Designate an Android Expert

There is greater divergence along operating system, device, and apps for Android than for either the iOS or BlackBerry ecosystems. Also, because Android is consumer-driven, the ecosystem changes rapidly, at consumer speed not traditional enterprise speed.

An essential best practice is to designate one individual on the Mobile IT team to be the Android expert. More and more of the overall IT team should gain Android familiarity, but our customers have found that they need one point person whose charter is keeping up with the rapid evolution of the Android ecosystem. Otherwise, Mobile IT's Android knowledge base quickly becomes obsolete. This person should be assigned before the company begins its Android pilot.

However, no one is an Android expert on day one. As a result, this individual must have several characteristics. He or she must thrive on reading Android blogs, testing new devices, and testing apps even outside work. He or she must participate in the Android Developer Community and have the people skills to credibly evangelize Android within the company. This individual is responsible for keeping the rest of the Mobile IT team up-to-date on Android and will become the trusted advisor for Android developers within the company. While it is tempting to rely on a third party for this role, it is best to make sure that the core knowledge is built and retained in-house. Third parties can absolutely provide assistance in designing overall strategy and developing and deploying apps, but if no one on the Mobile IT team is Android-savvy, future decisions will be ill-informed.

Assigning a point person to keep up with Android is the first step to devising an effective rollout strategy.

Step 2: Understand Your User Community

The push to Android will come from employees, not IT, so before developing a deployment strategy, IT must first understand what end users want and need:

- Which Android devices are users buying in their personal lives? From which carriers?
- Why do they select Android? Is it price, functionality, or experience?
- What do they like? Not like?
- What kinds of apps are they using on Android?
- Have they had or are they worried about malware?
- How do they compare Android to iOS?
- Do any of these preferences differ by form factor (smartphones vs. tablets)?

The answers to these questions will provide an understanding of the user expectations of Android within your company. If you begin a pilot with a device that the users have already rejected, the deployment will not be successful. The resulting knowledge may also not be easily applied to other devices in the Android ecosystem due to fragmentation. Also, if users are selecting Android purely for price but your company has a corporate-funded iOS deployment, then those same users may be perfectly okay with an iOS device the company gives them instead of an Android device they buy themselves. However, if user preference is based on factors other than price, or your company is planning BYOD programs, then this basic user research feeds the development of the initial short list for Android device selection.

Step 3: Determine Capability Requirements

The first section of this paper outlined the six core requirements for making Android enterprise-ready. Many IT leaders mistakenly assume that Android behaves similarly to Apple's iOS, and that is not the case. In addition, as discussed earlier, there are variances across Android devices themselves. Therefore, along with assigning an

Android expert, IT also has to determine the baseline capability requirements for certifying Android devices. Here is the direction we see many customers taking today to define their baseline requirements:

a) **Lost device security:** Remote lock, wipe, and password policy enforcement for mobile devices are a must for all companies. In BYOD initiatives, selective wipe to preserve personal data is equally important.
b) **Encryption**: Some companies will not accept Android until encryption is available across the entire device. Some are willing to deploy as long as e-mail is encrypted.
c) **Enterprise e-mail**: Most companies identify remote configurability of e-mail as a baseline requirement for deploying devices at scale. However, some IT teams are willing to support small Android deployments without remote e-mail configuration if user self-service is possible and intuitive.
d) **Secure connectivity**: Wi-Fi and VPN configuration can be confusing to end users. Most IT teams set remote configurability as a requirement when they allow mobile access to the enterprise beyond e-mail.
e) **Certificate storage**: Certificates serve the dual purpose of identity enforcement and user experience improvement. We see large companies moving down this road from the outset, specifically for e-mail, Wi-Fi, and VPN. Lack of certificate capability was a major factor in limiting Android adoption in large enterprises in 2011. However, several Android devices can now support certificates and we expect this support to be available broadly by end of 2012.
f) **Direct app installation**: Many companies are comfortable starting their Android deployments with just e-mail. However, due to the openness of Google Play, they still need visibility into app inventory from day one in order to protect against rogue apps. Internal enterprise app development for Android is still early stage at most companies, even though iOS app development might already be in full swing at those same companies. As a result, though app distribution and discovery through an enterprise app store is an essential Mobile IT requirement for Android, it can usually be added in later deployment phases.

Setting a baseline along these six capabilities will break through the initial confusion and establish the minimum level of enterprise functionality required for each phase of Android adoption. For example, remote lock/wipe, password enforcement and encryption may be sufficient for an Android pilot, though greater functionality would be required for a wider production deployment.

Step 4: Set the Support Boundary

IT must decide what to support, and what not to support. For Android, this might mean modifying mobile initiatives already in place for BlackBerry or iOS.

- **Device support boundary**: BYOD will likely be constrained to particular Android devices, instead of letting users bring in truly any Android device they have. The device support boundary will encompass a class of Android devices based on the enterprise capability requirements defined in Step 3.

- **App support boundary**: App monitoring is likely to be more important on Android than iOS, because Google Play is not curated as tightly as Apple's App Store. The presence of certain apps on a user's Android device may limit IT's willingness to support that particular device. However, some companies may still decide not to monitor app inventory, especially in BYOD scenarios with strict privacy policies.
- **Enterprise access support boundary**: Enterprise access is tied to trust. For example, an Android device without encryption has a low level of trust. An Android device with encryption, certificates, and appropriate policy has a high level of trust. Trusted Android devices will have access to all enterprise resources. Those at a lower level of trust will be blocked or allowed limited access to e-mail and apps.

These support boundaries must be clearly communicated to the user community in order to set expectations appropriately about the use of Android for work. Including these boundaries in the mobile user agreement is necessary for legal enforcement but not adequate for the end user. Mobile IT will need to develop supplemental forms of communication, like posters, alerts, and wikis, to educate employees on what is supported, what is not supported, and the simple logic underlying those support decisions.

Step 5: Plan a Phased Approach

A three-phased approach to Android adoption works best for most companies. They limit the possibilities at first and then add devices and functionality over time.

Pilot *(3-4 weeks):* Start small. Select a single device type to test with a pilot group of 5-10 users who are tech-savvy and 5-10 users who represent the general employee population. Deploy e-mail and one or two commercial apps. Manage and secure with a Mobile IT platform like MobileIron. After the pilot, assess the user experience. Was it buggy or intuitive? How many calls did users place to the help desk? How did it differ across the two populations? How complex were the issues? How effective was vendor support? Address any issues before expanding.

Pilot Plus *(3-4 weeks)***:** Add another device type from a second manufacturer. Add a tablet. Expand pilot size. Expand functional testing to include remote Wi-Fi and VPN configuration, certificate-based security, and additional functions like international roaming detection. Write a simple Android app, if possible, to understand the difference in development time and resources compared to other platforms. The goal is to gain the confidence and knowledge to deploy Android more broadly. Note that if internal app development is part of the Pilot Plus phase, it will extend beyond the target 3-4 weeks.

Production: The path to production deployment for Android will likely be both more controlled and more constrained initially than it was for iOS. But this final phase is also when the company can truly start to realize the potential of Android, as it is now more broadly offered to the employee base. At this point, deployment and management must be as automated as possible, with policies and technical infrastructure fully in place. This is where the preparation done in Steps 1-4 bears fruit.

Conclusion

Mobile IT departments that invest the time to learn both the advantages and challenges of Android—and proactively plan a path to adopt it—will be able to institute an inclusive mobile device policy that makes sense to users and drives their productivity and satisfaction.

Saying "no" to Android is not an option. Consumer adoption is exploding and best practice Mobile IT departments are conscious of the core mobile lesson of the past two years: the user wins. When iOS first hit the enterprise, most IT teams were not prepared and scrambled to educate themselves and set strategy. Android is more complicated, so it is even more important to be proactive instead of reactive in defining an approach.

Android enablement is a practical necessity but also a proving ground for Mobile IT's ability to adapt and flourish in the post-PC era.

Constant technology and behavior shifts are inevitable in a consumer-driven market. A structured methodology for Android will not only make specific Android projects more successful, but it will also prepare Mobile IT for the next big change, whether it is a new mobile operating system or a new class of user requirements. Android enablement is a practical necessity but also a proving ground for Mobile IT's ability to adapt and flourish in the post-PC era.