Chapter 2
Mobile Virtualization Technologies

In order to achieve increased adoption and sustainability "bring your own device" (BYOD) schemes within the enterprise, mobile virtualization is rapidly becoming a very attractive choice because it provides both employee and enterprise with flexibility while addressing the privacy concerns of the user and meeting the organizations security requirements. Allowing BYOD devices in the enterprise requires policies in place that govern how devices will be used and how they will be managed while maintaining end user flexibility. A number of technologies for mobile virtualization have been developed over the last few years which range from sophisticated mobile device policy management, to hypervisors and container based separation [JaKa01].

2.1 Mobile Virtualization via Device Management Policies

Mobile Separation can be achieved through the use of IT Security Policies that are managed by a Mobile Device Management (MDM) system. This type of approach is done by a server based management approach that lets the IT managers enforce policies across the user base which can be applied to the whole community or on a group basis so that it can be customized to the level of security required per group. For example, executives can have a stricter set of policies to include device encryption and a general group that allows for limited email, calendar and contacts support. There are many MDM's available now in the market and most of them support the major Operating Systems like iOS, Android, RIM and Windows. Some examples of these that provide a wide range of security policies that manage separation of applications and data are BlackBerry Enterprise Server, MobileIron and Good Technology amongst several more [Grum02].

D. Jaramillo et al., *Virtualization Techniques for Mobile Systems*,
Multimedia Systems and Applications, DOI 10.1007/978-3-319-05741-5_2,

2.1.1 BlackBerry Balance

Research in Motion (RIM) specifically took advantage of this marketing opportunity by focusing on the enterprise market and developed the BlackBerry smartphone along with a server component called the BlackBerry Enterprise Server (BES). RIM incorporated algorithms for managing the security of the device, applications and data in a secure and reliable manner. RIM further extended their technology and allowed the consumers to manage both, the personal and enterprise data and applications by introducing BlackBerry Balance technology solution. BlackBerry Balance extended its device security policies by managing them from the server and maintained a separation of enterprise applications and data from the rest of the system. The security layer clearly identified and elegantly separated the secured applications installed on the device with applications marked as unsecured. It further obstructed the user from exchanging data between secured and unsecured applications, like cut and paste, thereby preventing any information leakage [BaTe01].

Essentially, this solution allowed work-related data to be stored in such a way that it is not accessible by personal use applications; thereby, setting up for application classification of enterprise and personal data. For example, in the case of social network applications like Twitter or Facebook, BlackBerry Balance restricts the access to enterprise data from within social networking applications. From a personal use perspective, access to the personal phone is permitted when the BlackBerry smartphone is locked. An important aspect for the enterprise is the ability to manage the devices, especially when people leave the company. For this, the IT admin has the ability to either delete all device data or delete only the enterprise data and disassociate the device from the BlackBerry Enterprise Server [Epst01, Hale01].

BlackBerry Balance is included in RIM's latest version of BlackBerry Enterprise Server 5.0.3 and BlackBerry Enterprise Server Express 5.0.3 or later, which can be used for company owned or employee owned smartphones running BlackBerry OS 6, 7 and later [Hale01].

In summary, BlackBerry Balance leverages specific IT policies along with features built into the BlackBerry device operating system to provide the data and application separation for business and personal purposes. This gives the flexibility for the enterprise to support both consumer and corporate owned devices.

2.1.2 MobileIron Virtual Smartphone Platform

The MobileIron Virtual Smartphone Platform allows companies to manage multiple operating systems at a granular level, support corporate and employee-liable devices, enforce cost control, and create a private enterprise application storefront for employees [Mobi01]. By creating this storefront, MobileIron provides an effective private online app delivery system that is controlled via policies that controls

who can download or run the app. IT Administrator can also create and apply rules for application security, which can define the applications as required, allowed or disallowed [Cox1]. Through this policy control, MobileIron achieves the control and separation as to what applications can be installed or run on the mobile device.

2.2 Mobile Virtualization via Hypervisors

Virtualization via hypervisors gives the ability to run two or more instances of an operating system on the same phone, thereby giving the ability to run personal apps and services on one OS and the business services on the more secure OS. There are two types of virtualization approaches for this (Fig. 2.1).

Type 1—Bare metal virtualization
This type of hypervisor runs at the host mobile hardware level and has direct access to the hardware resources. This type of hypervisor can host multiple operating systems. Because there is direct control to the hardware, performance of each of the operating systems can be optimized. Furthermore, since each operating system is completely isolated from the other, this provides the best isolation and security from one another [CrSo01].

Type 2—Hosted virtualization
This type of hypervisor runs within the host mobile operating system environment, just above the hosted OS where the second software level runs and the guest operating systems run at the third level above the type 2 hypervisor—see figure below.

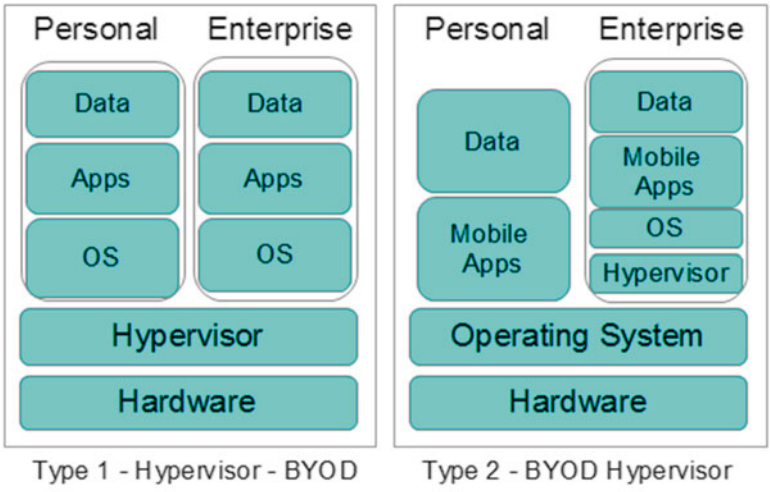

Fig. 2.1 Hypervisor types

Installation of the hypervisor is done on top of the guest OS because it is just like any other application. Performance of the guest OS is heavily dependent on the host OS. Furthermore, any compromise of the host OS will render the guest OS inoperative as well [CrSo01].

Hypervisor technologies have a downside in that they are required to work directly with OEMs which takes longer and there are fewer smartphones today that can support hardware level virtualization [Bran01], however, this will change over time as the ARM Cortex-A7, A15 and similar processors are incorporated into more mobile smartphones/tablets and also as standards like Virtualization Management Object (VirMO) proposed by Red Bend in the Open Mobile Alliance Device Management (OMA DM) Working Group [Red01].

2.2.1 KVM on ARM

Kernel based Virtual Machine (KVM) is a virtualization infrastructure for the Linux kernel that supports native virtualizations on processors with hardware virtualization extensions. KVM/ARM is a virtualization solution for ARM processor based devices that can run virtual machines with nearly unmodified operating systems. Since the ARM CPU processor is not virtualizable, KVM/ARM uses a lightweight paravirtualization [DaNi01] via a script-based method to automatically modify the source code of an operating system kernel to allow it to run in a virtual machine. This lightweight paravirtualization is architecture specific but operating system independent as seen on the above figure [DaNi01]. These changes in the guest OS kernel are made so that it can take care of sensitive non-privileged instructions (Fig. 2.2) by doing the trap and emulate methods which are then handled by an interrupt handler which then emulates the appropriate functionality [Rama01].

2.2.2 Xen Hypervisor on ARM

Xen is an open-source hypervisor that allows for multiple operating systems to safely share the hardware via resource management without sacrificing performance or functionality [BaDr01]. Figure 2.3 illustrates a basic Xen configuration where the hypervisor consists of a small layer on top of the physical hardware. It implements virtual resources such as vMemory, vCPU, event channels and shared memory, and it controls the assignment of I/O devices to VMs. The user domains—DomUs are started by the Dom0 and they can run any paravirtualized operating systems like Linux and others. These guest OSs have minimal changes where privileged operations are changed to calls to the hypervisor [SaVa01].

Xen has been ported to the ARM architecture [Xen01] used for secure mobile phones supporting enhanced security features for mobile devices with mandatory

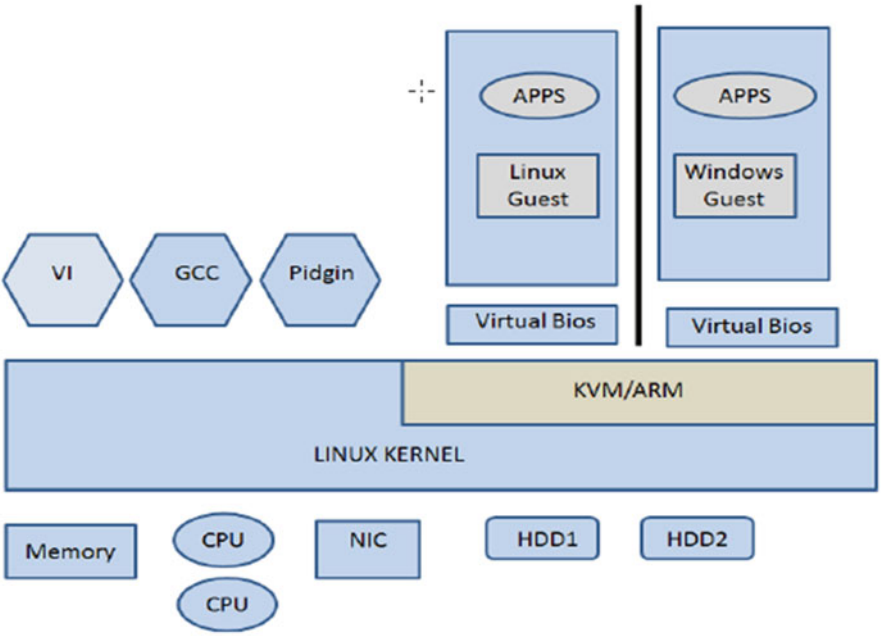

Fig. 2.2 KVM overview [Rama01]

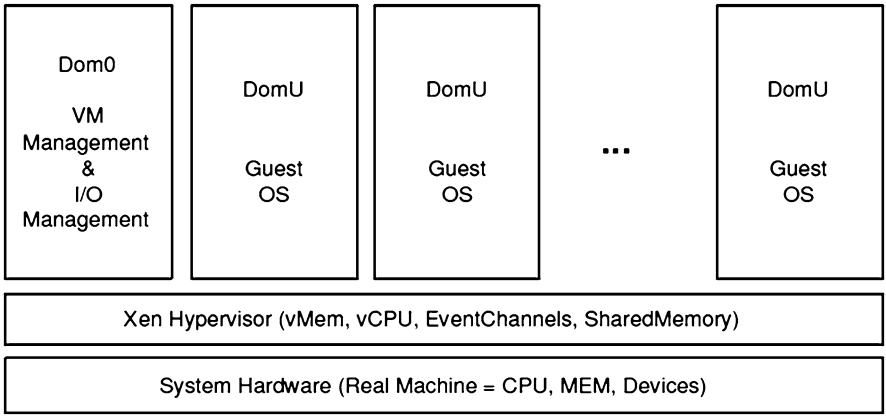

Fig. 2.3 Xen hypervisor [Xen01]

access control through access control models and a secure boot process which is designed to detect any alterations of the VM during the bootstrap process. Samsung has taken a keen interest in this project and has developed a version that supports ARMv5, ARMv6 and ARMv7 processors, the later one supporting the new virtualization extensions [Morg01].

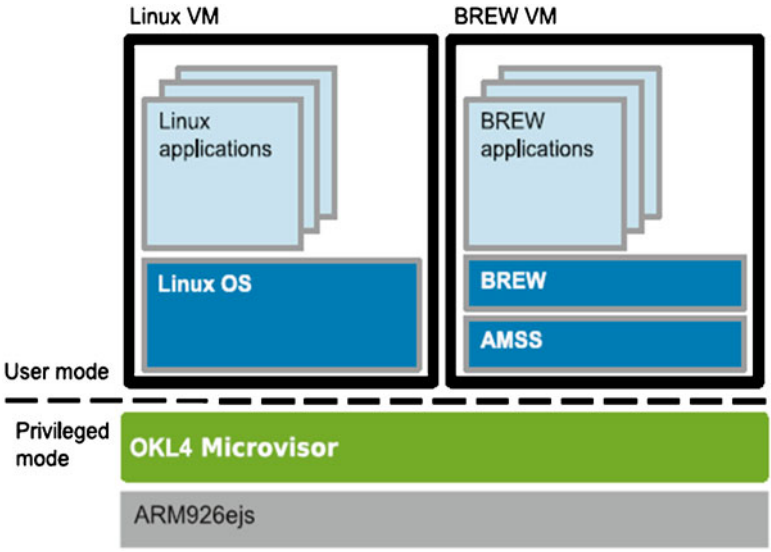

Fig. 2.4 Evoke software architecture [Heis01]

2.2.3 OKL4 Microvisor: Open Kernel Labs

Open Kernel Labs (OK Labs) is a provider of virtualization software for mobile devices, consumer electronics and embedded systems. Its leading offering is the OKL4 Microvisor which has been embedded into more than 1.2 billion devices, including almost all CDMA phones because of the strong partnership with Qualcom [CrSo01]. The Okl4 Microvisor is a type 1 hypervisor that can be either built into the device at the OEM level or applied after the fact via OK Lab's Virtualization Over the Air (VOTA) process, which is similar in concept to Over the Air (OTA) firmware updates.

2.2.4 Motorola Evoke AQ4

The Motorola Evoke was the world's first mobile phone that uses hypervisor virtualization, implemented using OKL4 as the core virtualization technology. The requirements for the phone were such that they only could be met by a design based on virtualization. This was accomplished by using a phone with a specific price point based on a single core design using an ARM9 core, a user interface running on Linux (OS), the baseband stack running outside of Linux, and components from BREW UI framework were re-used (Fig. 2.4).

Evoke's software architecture is shown in the diagram below, where there are two virtual machines running on top of the OKL4 Microvisor which interact via the OKL4 message-passing IPC as well as through shared memory. The complete

Linux system is running de-privileged and the AMSS/BREW baseband stack/OS are both running in user mode. Due to the high performance of the OKL4 Microvisor, the virtualization overhead was kept to almost unnoticeable levels which in many respects were better than what was achievable with running native Linux.

As a result, Motorola produced an attractive device with a snappy user interface that was even more responsive than other non-virtualized phones, even those which were based on more powerful ARM11 processors [Heis01].

2.2.5 VMWare

VMware is committed to bringing virtualization to the mobile handset and two manufacturers—LG and Samsung have announced their support for this solution on Google Android [Thom01]. At VMworld 2011, VMWare announced VMware Horizon Mobile Manager, previously known as VMware Mobile Virtualization Platform (MVP) which allows Android phones to use virtual machine technology to run a second instance of Android, very much the same way virtualization works on servers and desktops. This solution basically has two separate phones running on one device, and can switch from the personal one to the corporate one by clicking a "work phone" icon. By isolating the employee's work environment from their personal environment and providing IT managers a Web-based management console to control what employees can do on the work portion of the phone, the user can have a more relaxed personal experience while the enterprise can have a manageable and secure environment [VMTN1] (Fig. 2.5).

Having two environments on the same device doesn't really make things more complex for the user because common functions such as receiving a call are active regardless of which environment is active. VMware says performance impact will be minimal and that the offering will work on both single and dual-core processors. Google Android was picked for the development of this platform largely due to the flexibility of it being open source. Initially, LG signed on to the project last December and now recently, Samsung has joined in providing a wider variety of devices such as the Galaxy S II phones and Galaxy tablets. In the future, more devices and other manufacturers are to be supported according to press announcements [Zieg01].

It is possible today to do many work activities on Android phones and tablets, but the real issue is how to ensure that the enterprise doesn't get affected by malware, viruses or losing data when the device is lost. This is why VMware's Horizon Mobile is very attractive because the enterprise phone is not affected by any malicious software and can be managed by IT administrators. If the device is lost, it can be remotely locked or wiped. Furthermore, the virtual phone can be provisioned with standardized or custom templates and also push out application updates over the air. User policies can be defined that can restrict what functions/features can be used and configure security features such as device lockup timeouts and passwords. From the employee's perspective, they are happy because they can now use their favorite personal device to do their work as well as use their favorite personal applications without having to carry two different devices [Whit01].

Fig. 2.5 MVP personal/enterprise screen shots [VMTN1]

for a physical in-person installation. Furthermore, the IT administrator can make adjustments, add/remove applications, push down new templates without having to reload the device. (4) Review health of the deployment and vital stats from a dashboard. (5) Application management with the ability to push applications over-the-air to the work phone from the application catalog. Allows the add/remove of apps and dynamically pushing the updates to the device. Multiple application versions are allowed. (6) Lock or wipe to de-provision the work phone which allows the ability to lock or wipe a device when the device is lost or sometimes the user might want to just reload the enterprise side [VMTN1].

On the device side, VMware Horizon Mobile Platform is built around a type 2 mobile hypervisor that is based on a lightweight paravirtualization technique for ARMv7 cores that is aimed at minimizing the total system complexity. A series of device and platform virtualization approaches for storage, networking and telephony are implemented which are key in enabling the performance, reliability and security of the system. Finally, this hypervisor is applied to the virtualization of the Android operating system, allowing it to run both the guest and host environments [BaBu01].

VHMM differentiates from previous approaches of system virtualization on the ARMv4-7 architectures that have entailed some form of core paravirtualization like Xen on Arm [SaVa01] by employing a distinct shallow paravirtualization approach that requires only the identification and replacement of sensitive instructions [BaBu01].

2.2.6 Red Bend: vLogix Mobile

Red Bend acquired VirtualLogix, a provider of mobile device virtualization solutions which it delivered to semiconductor vendors, OEMs, ODMs' service providers, and systems integrators. Over one million devices shipped with VirtualLogix's type

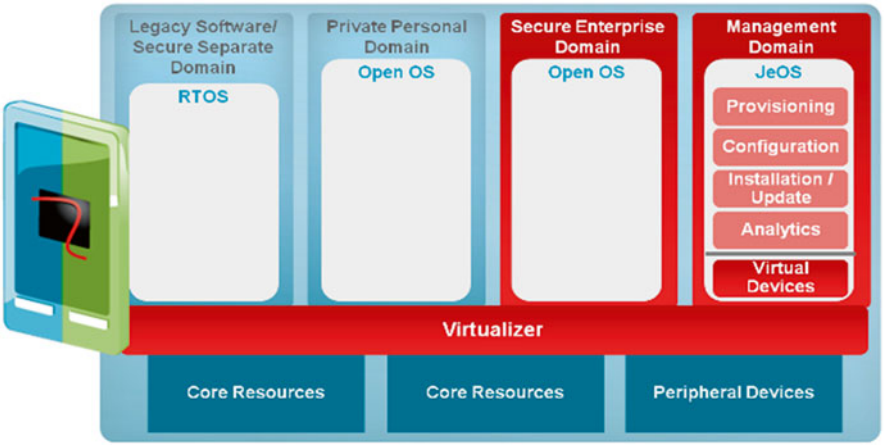

Fig. 2.6 vLogix software architecture [CrSo01]

1 hypervisor technology embedded on devices like the Acer beTouch E110, E120, and ch E130 models; HTC Tianyi; KTouch W606; and CoolPad Yulong W711. Red Bend adopted this technology to fit into it's broader portfolio and also saw it could use its strength, which is its ability to partition secure software domains that can be managed separately [CrSo01].

This offering supports processors based on the ARM Cotex-A15 and Cortex-A7 cores in single and multi-core configurations. Red Bend is enabling device manufacturers to take advantage of this latest family of cores without the need to modify an existing high-level operating system (HLOS). vLogix Mobile performs management of the chipset's multi-core architecture in the Virtualizer, allowing the HLOS to remain as is. Users of this technology benefit by not having to redesign, redevelop and revalidate existing software to support new OS configurations [Red01] (Fig. 2.6).

The solution is comprised of the Virtualizer which is a type-1 hypervisor that runs together with suite of software modules that are configurable according to the desired deployment. The Virtualizer runs on the host hardware, also known as a Bare Metal Hyper, and schedules access to the shared hardware services like file systems, serial lines and network interfaces amongst the virtualized operating environments. System resources like RAM and persistent storage like flash memory are partitioned and allocated according to the performance demands of each of the operating domains or virtual machines. Hardware resources like CPU, clock and memory management unit are also virtualized for each of the guest operating systems and the access to the actual hardware is allocated by the Virtualizer [Red01].

Virtual devices are provided for each of the guest operating systems that access system resources like screen, 2D/3D hardware acceleration, multimedia acceleration, Wi-Fi, GPS and other input/output devices. This provides a secure separation in each of the OS/domains to the features of each device without the physical access to the device itself. The Domains are virtual machines that are designed to run

virtualized images of Android where IT administrators deploy over the air a secure enterprise environment that can run corporate applications and provide network access on a consumer owned device [Red01].

Red Bend's solution is simple to use, where the user clicks on an icon on their home screen that takes them to work image that is running their business apps and easily switches back to their personal side by clicking on the personal home icon. IT administrators can control what applications are installed on the enterprise side and also exclude services like Android Market/Google Play [Gohr01].

Notifications in the business partition will appear on the user's notification bar where they can then go back to the home screen and switch to the business partition. The difference between Type 1 and Type 2 hypervisors can be visualized by noting the change from one platform to another when one is done at the phone lock screen where the other one is done at the application level, thereby providing an increased hardware level security and better performance [Bran01].

2.2.7 Cells

Cells' virtualization architecture is used for enabling multiples smartphones that run simultaneously on the same physical cell phone in an isolated manner. This architecture follows a usage model where there is one foreground virtual phone and multiple background virtual phones. A device namespace mechanism and device proxies are integrated with a lightweight operating system virtualization to multiplex phone hardware across multiple virtual phones while providing native hardware device performance. The platform includes a fully accelerated 3D graphics, complete power management features, and full telephony functionality with separately assignable telephone numbers and caller ID support. A prototype was implemented of Cells that supports multiple Android virtual phones on the same phone. Performance results demonstrated that Cells imposed only modest runtime and memory overhead, worked seamlessly across multiple hardware android smartphone devices and transparently runs Android applications at native speed without any modifications [AnDa01].

2.2.8 Cellrox

Cellrox is an Android based technology that has roots on the Cells project [Cell01] and it is not considered to be a complete OS virtualization because the virtualization is limited to the user space, creating multiple personas that share a common Linux kernel. Since the kernel is shared, all of the personas have to be a similar version of Android. Cellrox provides the same look and feel as the original operating system in order to provide a common user experience between the personal and work personas. Multi-tasking between the various personas works much the same way as

how multi-tasking works between different Android apps where only one persona is in the foreground at any given time and the apps running in other personas run just like any other background app. Cellrox has the ability to support more than two personas which gives the ability to have a multi-tear level of locking down applications. For example, the personal persona can be very relaxed, while a work persona has semi-restricted environment with standard security policies for the enterprise and third persona could be a completely locked down environment for highly confidential applications. One of the outstanding features that Cellrox features is the ability to have shortcuts of the applications from the different personas such that the user doesn't have to jump between the different personas to find the application which is the case in most of the separation technologies [Madd02].

Usability is a key component in Cellrox. Only one persona is in the foreground while the others run in the background. The user can easily switch between the personas by using a custom key-combination to cycle through the personas or by swiping up and down on the home screen of a persona and each persona has an application icon that can be launched to see a complete list of available personas to select from. For security purposes, the system can be configured such that a no-auto-switch option will prevent background personas from being switched to the foreground without explicit user consent, preventing a background persona from appearing unexpectedly. An auto-lock feature can be enabled that will require the user to unlock the persona using a pass code or gesture whenever a persona switches from the background to the foreground. CellRox's technology was evaluated through an experimental study at Columbia University and the results demonstrated performance benefits in their approach, suggesting no noticeable performance difference between the operation of the mobile device in a persona compared to the native device operation [Cell02].

2.3 Mobile Separation via Containers

Another approach to providing separation on a mobile device is via Application Containers. This is achieved by using a solution more like Unix method of multiple users where you have one box with multiple users logged in and each user has their own experience. Each user has its own experience and all users run concurrently with one kernel and one operating system [Cree01]. Applying this logic, one user is the personal side and the second user is the enterprise side.

2.3.1 Good Dynamics Technology

Good Technology provides two technologies that address BYOD. First is Good for Enterprise—a secure e-mail, mobile device management and a Intranet-Internet proxy server solution targeted for enterprises. Second is Good Dynamics platform

that brings necessary tools, infrastructure, and APIs to developers, allowing them to provide secure applications across devices and operating systems. This protection is delivered by containerizing data at the application level which is accomplished by wrapping a layer of protection around the enterprise deployed apps, which separates the corporate data from the employee's private information and consumer applications. Through this containerized approach, Good Dynamics establishes a secure application environment that minimizes the possibility of data loss. The containerized applications provide the employee the freedom to access enterprise data in a safe and secure manner while being able to switch back and forth with their personal applications without compromising company information. This container-based method applies secure and encrypted transmission of data end to end, from the enterprise servers behind the firewall all the way to the mobile device. Good Dynamics architecture requires that every application use the Good Dynamics APIs and be compiled/linked with its SDK in order for it to run within its secured container. This limits the number of commercial applications that can run within this container.

2.3.2 Divide by Enterproid

Enterproid introduced a mobile virtualization solution in late 2011 called the Divide Platform which gives users a way to use their smartphones for both work and personal life. This solution which runs on Google's Android devices 2.2 or later and Apple's iOS devices like iPhone and iPad is designed to provide multiple profile support, a set of productivity apps, as well as a personal and enterprise cloud management system. The Divide system truly blends in well providing the end user true separation from the work environment without compromising personal freedom [Dolc01].

Divide functions as a container application, with security policies and management features applied solely around that application, leaving the rest of the user's device untouched. Organizations can then in turn manage their own applications within the container. The management features are essentially the same as with most MDM solutions: password policy, encryption, data isolation, clipboard restrictions, remote wipe, screen locking and more [Madd01].

Virtualization is done by having separate secure profiles for work and personal environments. The personal side enjoys the freedom of all the functions, features and applications including access to the Android Market, where as the work side is managed by the IT administrators with a separate more strict profile suited for the appropriate enterprise. Switching between the two environments is done by a simple double tap of the Home key or can also be done via application icons on each of the sides or by going to the notification/alerts bar (Fig. 2.7). Separation of applications is essential in a virtualization system because you want to avoid data leakage from enterprise into personal applications. All applications in the container side

Fig. 2.7 Divide personal/enterprise screen shots

benefit from an encrypted 256 bit storage. Furthermore, encryption is not dependent on the OS, hence not compromised immediately on rooted/jailbroken devices—all the encryption is built within the application. Another example of separation is to limit the potential for leaking enterprise information into personal social networks by limiting the transfer of information between the personal side and enterprise container by restricting the ability to cut/paste information between the two environments. Additionally, Divide provides a rich set of native-Android office applications like mail, calendar, tasks and contacts. The mail application provides threaded email conversations which can also be searched both on the device and on the server. The enterprise mail, calendar, contacts are configured and delivered via the standard Active-Sync API's which provides the ability to connect to various email backend systems like Google, Yahoo, Microsoft, IBM Lotus Notes, etc., All of which are fully synced via 3G or WiFi networks, thereby providing ultimate flexibility wherever you are at. Most importantly, because all these applications run in the separate work environment, everything is fully encrypted and compliant to the policies defined by your IT administrator [Haza01].

Another set of features that furthers capabilities of the system is a cloud based management portal that allows the user and the IT administrator to manage their devices. The user portal, also referred to as "My Divide," provides the user of the smartphone a number of features to control their device, such as the typical device wipe, reset both device and divide password, lock device, but it also provides features like just wiping enterprise data, audio beacon to locate device, device location, push a URL do the device browser and more. The portal also provides additional tabs that give the user the ability to view their network usage, the applications installed as well as a detailed list of the state of the device components like the phone, WiFi, battery, network, audio, location and more.

The IT admin portal, also called "Divide Manager," provides some similar functions such as location, device or enterprise wipe, and password reset, but it also

functions where the system is configured by defining groups of users, device policies and enterprise applications. The device policies allow for configuring the typical mobile device management settings, but what is different here is that these settings are for managing the enterprise partition. Here you can set the device password quality, length, expiration, history and lockup timeouts.

Security features allow for controlling clipboard sharing, actions taken when a user removes their SIM card and even checks to see if the device is rooted and gives options on what actions to take if this were to take place. Another key piece of the system is the ability to manage applications in the enterprise partition. The admin is able to upload specific applications that will get pushed to the device upon installation of Divide. Furthermore, the system allows for the ability to control what apps are allowed or not allowed to be installed in the enterprise partition. System performance and battery life are well maintained because these are managed by the operating system and does not require direct access to the hardware [Cree01].

Divide differentiates itself from other solutions is that it runs as an application and it does not require any cooperation with the phone OEM. The install does not require any low-level drivers and uses the standard Android procedures for installing applications. It is a light weight solution that shares much of the device resources and significantly reduces the device overhead required by virtualization and also delivers 256 bit encryption for data [Bran01].

Overall, Divide demonstrates a secure and flexible system for enterprises to use, especially for their BYOD community. It is very functional and provides a wide range of security options and features as well as giving the end user the freedom of their device on the personal side.

2.3.3 TrustDroid

TrustDroid is a practical and lightweight domain isolation solution that runs on the Android OS. It provides application and data isolation by controlling the main communication channels in Android, mainly IPC (Inter-Process Communication), files, databases and, socket connections. This solution is lightweight because it has a low computational overhead and does not require duplication of Android's middleware and kernel like other virtualized solutions. It also organizes applications along with their data into logical parallel domains. Figure 2.8 illustrates different methods for achieving isolation. TrustDroid is shown in Fig. 2.8a where it extends Android's middle ware and kernel with mandatory access control. OS-level virtualization is shown in Fig. 2.8b and this is typically seen in Application level containers. Hypervisor based technologies is shown in Fig. 2.8c. The areas designated in black are the trust computing base (TCB) which is responsible for the security enforcement on the platform and also trusted by the enterprise. TrustDroid has the largest TCB, however, it is one of the most lightweight because it doesn't duplicate any portion of the operating system stack and provides good isolation.

At runtime, all application communications are monitored, as well as access to common shared databases, file-system, networking, and denies any data exchange

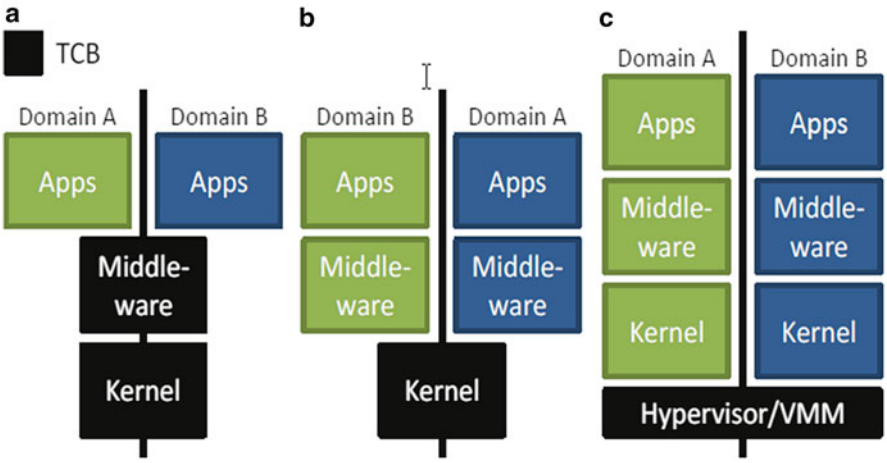

Fig. 2.8 Approaches to isolation [BuDa01]

or application communication between different domains. TrustDroid adds a negligible runtime overhead and compared to other virtualization approaches, only minimally affects performance and battery life [BuDa01].

2.3.4 Android 4.2: Multi-User

Google in its latest version of Android 4.2 OS has the option to have multi-user support which gives the device owner provide separation across users [Cabe01]. This becomes a very interesting development because it gives the device owner the ability to have it's own version of separation—one user for personal use and a second user for enterprise use. It will be interesting to see if enterprises will leverage this functionality in such a way to accommodate and attract the BYOD community to use their personal device for enterprise use as well in this type of environment. In order to achieve this, it will be necessary for the enterprise to deploy its device management software and policies on the enterprise side while allowing the personal freedom on the personal side. This will provide the best experience for the personal side while preserving the security and separation of the enterprise side.

2.4 Summary

The explosion of mobile devices in the consumer space has been quite a disruptor for the mobile enterprise where corporate managed platforms are well defined, well contained and fairly secure. However, now with the emergence of the consumer mobile devices from Apple iOS and Google Android, the enterprise has been forced

to make functional tradeoffs in order to maintain platform and data security. Due to the corporate security decisions, the end user gives up a significant amount of personal freedom and ease of use of their device. Enterprise security requirements like password complexity, device encryption, network restrictions and other techniques that restrict the access to information on the mobile device tends to drive users away and/or encourages users to find other less secure alternatives that will eventually compromise enterprise data and access. A number of mobile virtualization technologies have been presented here, each of them delivering specific features that focus on ensuring the security of the enterprise container and applications. Application level containers, type 1 and 2 hypervisors all lack the organic integration of enterprise and personal personas found in solutions like the emerging BlackBerry 10 Balance platform. Hypervisor technologies show promise; however, it is specific to the Android platform, has higher end hardware requirements and is not likely to be seen on the iOS platform anytime soon.

http://www.springer.com/978-3-319-05740-8

Virtualization Techniques for Mobile Systems

Jaramillo, D.; Furht, B.; Agarwal, A.

2014, XIV, 73 p. 38 illus., Hardcover
ISBN: 978-3-319-05740-8