

Enabling Your Mainframe Data on Mobile Devices

Analyst: Mike Kahn

Introduction

Whether you drive a four-wheel drive pickup truck, an all-wheel drive SUV, a seven-passenger minivan, or a two-seater sports car, you have chosen your vehicle to meet a specific set of needs that you may have prioritized and for which compromises may have been made. In fact, you may have more than one vehicle – say an SUV and a sports car – because you want to have the right vehicle for your very different needs (or seasons). The same is true for the computers and storage on which your business relies. You probably have multiple tiers of servers and storage, conveniently arranged to meet your IT requirements, some mission-critical, some business-critical, and some less critical.

If your enterprise has a mainframe (or several), the reasons primarily tend to ensure that your enterprise's most critical transaction processing, information management, and data analyses get done with a very high-degree of reliability, data protection, and performance – especially at high volumes. These days, mainframes are the workhorse that delivers what other architectures only can hope to achieve – exceptionally high qualities of service, reliability, and integrity for very high volumes of business transactions and data management. While we could debate this point forever, if you have critical work being done on a mainframe, let's assume that you know that this is a really good thing. This paper is for those who need to access or deliver that mainframe data to or from mobile devices.

So what does it mean to be in this mainframe-centric camp? It means that you have much, if not most, of your most valuable data “sitting on a mainframe”. That has been a good thing and is a primary reason that you keep investing in your mainframe environment. However, critics might say, “Hey, your best data is trapped on that mainframe.” This has been said for many years and many data centers have chosen to “liberate” that data by copying it to other server environments and setting up a number of data warehouses and/or data marts to meet the real-time access and analytics requirements.¹ Of course, there are two (or more) opinions on whether this was a good thing to do, primarily because of the high costs of duplicating (often many times) the amount of data stored and the unproductive reality that now the copied data is no longer up-to-date (unless it is stagnant or historical). If your mainframe is processing tens of thousands of transactions per hour, minute, or second, then it is hard to keep an offloaded data collection consistently up-to-date and this may just be what you wish that you could do.

In the old days, which might be seen as recently as 20 years ago, access to data was constrained to those who had a “real need”. So isolating data on any server was not necessarily a bad thing. Working on a data mart with week-old data was considered state-of-the-art. Of course, the world has changed dramatically. **The pace of business and decision-making is now being done in real time.** Operating on stale data is a quick path to failure

IN THIS ISSUE

➤ Setting the Stage	2
➤ Looking “Under the Hood” – A Further Examination of the Requirements	4
➤ IBM’s Answers	10
➤ Conclusion	16

¹ Many reasons have been given for this “offloading” decision, including (a) the cost is perceived to be less for the outboard servers; (b) adding capacity is easier with outboard servers; (c) the software is better on those outboard servers; (d) and it is easier to do what is desired on those outboard servers. If any of those arguments were true (in the past), they are not true today. However, a detailed discussion of these arguments only is tangentially related to the topic of this paper, and they only will be discussed secondarily.

in today's highly competitive, "always-on" world. **Meeting the needs for real-time access to critical data is faced with new, very important business and IT challenges in an era of mobile devices, such as smartphones and tablets.** Even if you have some doubts, accept as a given that little is more important to most businesses than having a positive, hopefully revenue-enhancing, mobile presence for employees, partners and customers.

The ante has been upped and mobile interactivity is the price of entry to the most important game in town, and around the world. Today, mobile is a primary force and may become your brand, if not a key to corporate survival. Data from activity (informational, social, and transactional) on mobile devices is rich with new insights and new opportunities, and the user is king. Activities in the *world of desktop*² are different than *world of mobile*, in the following ways.

- *Sit back and read to read while moving*
- *Document oriented to message oriented*
- *Large, complex apps to purpose-built mini-apps*
- *Context neutral to context aware*
- *Task driven to notice driven*
- *AC powered to battery powered*
- *Predictable network response to unpredictable network response*

From this list alone, you should be able to conclude that while some of the uses may be the same or similar to what was done on desktops, many uses will be dramatically different.

Additionally, with mobile enabled, the informational needs may need to go in both directions³. Information from mobile devices is valuable to business processes, especially "instantaneous" marketing and decision-making. Good examples are the GPS and camera on each smart device. Knowing where your user, customer, or prospect is located geographically might be key to what you deliver, propose, or suggest to them. The same is true for what each user might be seeing (or hearing), as might be captured by the camera (or microphone) in a smart device⁴. **Enterprises need to integrate the most up-to-date information in critical data collections (on the mainframe) with what is being "sensed" locally on the smart**

devices⁵. If your customer is physically in your retail store and surfing your website, you need to know this and to interact accordingly. This is even more important, if the mobile user is in your competitor's store while surfing on your site.

There may not be a singular (one-to-one) connection between back office intelligence and a single smart device. In an era of social media, there may be situations where your applications want to know the relative proximity of a large number of your employees, customers, and/or other interested parties and the application may wish to communicate *en mass* to all of them or a data-driven subset of them.

The question for this paper is whether you need to provide secure, timely and speedy access to your critical data sitting on your mainframe from that now-very-important mobile and diverse world of smart devices – and vice versa. If this is a hot question for you, read on to learn how this might be done in a forward-looking way. If not, enjoy the blissful world in which you operate...for as long as you can.

Setting the Stage

Mainframe transaction processing systems (CICS⁶) and data management products (DB2, IMS, and VSAM) are involved with most of the data stored on a mainframe. Together, these are central for the creation (usually by some form of transaction processing), organizing, storing, accessing, and analyzing of critical data. Without question, there are many other pieces of middleware that might be involved, some of which will be discussed later. Nonetheless, **integrating smart devices with CICS transaction processing and with the critical mainframe databases is the primary challenge** and, thus, the focus of this bulletin. If you understand how to do this, then it will be much easier to extrapolate the approaches and methodologies to other mainframe processes and data.

For many, this might not seem much different from the old days when relatively dumb green-

² Desktops include laptops, generally anything that might have been labeled a "PC".

³ From central data store (on the mainframe) to mobile user's smart device and vice versa.

⁴ Given all of the current excitement for the first generation of *Google Glass* (camera-enabled glasses), you can imagine the possibilities ahead.

⁵ In this paper, the phrase *smart devices* is used interchangeably with *mobile devices*, but the former may be the more descriptive moniker, mostly because it is their local intelligence that makes them an interesting IT endpoint. Of course, being mobile is an assumed characteristic of many smart devices, i.e., smartphones and tablets, but there are other smart endpoints to consider, such as remote data capture devices. In many of the IBM-sourced exhibits, *mobile devices* is the moniker used. All refer to the same devices.

⁶ CICS=Customer Information Control System, which runs on z/OS. It has been at the heart of mainframe transaction processing for decades.

screen terminals gave way to added interfaces on increasingly more sophisticated terminals and personal computers. However, there several are important differences. **In the old days, we were content with just being able to present mainframe data and processes to a relatively uniform collection of outboard devices.** Yes, we had many variations, but the *Windows*, *MacOS*, and (later) *Linux* operating systems were the commonizing elements for these “personal” devices, both stationary and portable. Applications written for each of these generally would run on multiple generations of personal computers from a variety of hardware vendors.⁷ So we went from “screen scraping” (retrofitting old application presentation layers for more modern uses) to standard web protocols for executing over the Internet (like SOAP⁸). Robust development and application solution sets were developed to access mainframe data and continue to be enhanced. IBM’s *WebSphere Application Server* was the vehicle for much of this. It operates on all of IBM’s systems platforms, including the mainframe, *zEnterprise*⁹, so you can write the application code once and run it on your choice of server platform(s). This was and still is important, but **more is needed today, in an era of ubiquitous smart mobile devices.**

Smart devices have changed the rules and the IT requirements, for the following reasons.

- Often they are shared
- Often they have multiple personas and uses, mixing personal and business
- Often they are diverse in their characteristics and capabilities
- Often they are used in many locations
- Often they allow the user to set the look and feel and prioritize how it is used

As a result, smart devices present many unique security and management problems.

Widespread use of smart devices now defines (broadly) a new ubiquitous interface. While looking down from 20,000 feet, we see a handful of smart device operating environments (e.g., *iOS*, *Android*, *Windows*, and *BlackBerry*¹⁰). However, on closer examination, **we might see hundreds or even thousands of smart device variations.** Each

cell phone and tablet is generationally-defined (e.g., Google’s *Gingerbread*, *Honeycomb*, *Ice Cream* and *Jelly Bean* versions of *Android*, for example). That makes it hard enough, but it gets harder, as there are many releases of each. And it gets worse, much worse, when you look at the vast number of hardware offerings (smart phones and tablets) that come from a large number of providers. Even Apple’s *iPhones* and *iPads* are delivered with some individuality, depending on which communications provider (like AT&T, Sprint, T-Mobile, and Verizon in the U.S.) is delivering the smart device. More importantly, even seemingly similar physical devices may have different physical and operational characteristics. For example, *what is the screen size and pixel density? How much real estate is available to an application? What kind of touch sensitivity is provided? Is there a stylus? How is security handled? Is the connection fast or slow, or steady or varying?* These result in a vast number of smart device “platforms” (unique combinations of operating environment, release, hardware characteristics, and provider “enhancements”) with which to deal. **What we have here is, almost certainly, a one-to-very-many relationship – hopefully (at best), one relatively well-controlled and generationally compatible set of applications, middleware, hardware and data (on the mainframe) needing to communicate and work with very many somewhat-unique smart devices.**

Enterprise data processing (and forgive the use of what some consider a seemingly old phrase, but it is most appropriate here) demands a higher quality of service than more mundane application environments, like what may seem to be the norm on your personal computer or, especially, on your smart device. That is why there is so much testing of enterprise-class hardware and software before introduction to widespread use by those inside and external to the enterprise. **Thus, on one hand in the data center we have a near-maniacal focus on “being ready for prime time” (with all of the critical issues addressed), while on the other hand we have thousands of smart device environments running on millions of smart devices where “instant satisfaction” seems to be the universal expectation, as in “if it doesn’t meet my needs right away, I will move on and try something else.”**¹¹

The purveyors of smart device operating

⁷ With Apple being a notable exception to multiple vendors.

⁸ *Simple Object Access Protocol*, which is XML based.

⁹ For a good summary on *zEnterprise*, see **The Clipper Group Navigator** dated August 28, 2012, entitled *The IBM zEnterprise EC12 - Bigger, Better, Faster* and available at <http://www.clipper.com/research/TCG2012019.pdf>.

¹⁰ To mention the players with the most visible presence, but there are many more.

¹¹ Another common expectation is “I can’t wait until the next version/release/gadget.” Change is the norm and comes frequently!

environments (Apple, Google, Microsoft and BlackBerry, to name just a few) each strive to put a unique experience in the hands of the users. While many have the similar gross functionalities, each is noticeably different. Each tends to be on a cycle of improvements (releases) of once or twice a year. And the purveyors of the hardware (like Samsung, LG, Nokia, Sony, Amazon, and many dozens more) all have their own “swizzles”¹², further compounded by the mobile device carriers and their additional swizzles and their own schedules. The key point is that end users have grown accustomed to the familiar (“swizzled”) environment that they use and customize further with apps, preferences, etc. While you might be able to generalize about a given release of Android (*Jelly Bean*, as an example), it is clear that the delivered experience is different when it comes down to a Google *Nexus* device versus a seemingly similar one from Samsung. **And it is this customized and personal experience that the end users come to covet.** This is why each of the operating environment vendors is trying very hard to create an ecosystem that captures the end user. The proof of the success is from Apple, where iPhone users are most likely to choose an iPad as their tablet, *iCloud* for storage, *iTunes* for their music and media, etc., but there are many proof points of this. Thus, there is much more than a bit of the “Wild West” here, with users taking “sides”, akin to the late 19th Century feuding between the *Hatfields and McCoys*¹³. **Now, there is a taking of sides, in general terms by operating environment, but also by the specific variation that has been customized and now is familiar to each end user.**

What users want to preserve is their personal experience with their chosen smart device(s). This results in fierce independence, especially for the users bringing their own devices to the enterprise playing field.¹⁴ Not only will it be harder to corral these users into doing things “the enterprise way” but presenting a total different experience than the user has on their personal devices

¹² *Swizzle* – derived from a “swizzle stick”, a device for stirring a mixed drink, but more loosely “a concocted mixture”, in this case a specific variation of smartphone technologies (hardware and software).

¹³ See *Wikipedia*, Hatfield–McCoy feud (as of Mar. 31, 2013), http://en.wikipedia.org/wiki/Hatfield%E2%80%93McCoy_feud.

¹⁴ *BYOD (Bring Your Own Device)* is a blessing and a curse. It is a blessing in that many enterprises are relying on their employees, customers, et al, to pay for their smart devices (instead of the enterprise). While this may be a capital-saving benefit, it tends to be more than offset by the lack of control and standardization that personally-owned devices provide to the enterprise’s IT organizations.

may be productivity-reducing. Thus, you have a very big challenge, with many, many devices and users who see themselves as the ruler of their personal smart devices. In addition, these are devices that often get lost or stolen and whose users may approach them quite casually.

Thus, the stage is set. You, representing enterprise expectations, want to expose your best and most precious data and processes into a chaotic, diverse, and generally untrustworthy world of many smart devices and end users’ personalized preferences and “good-enough” expectations¹⁵. *How can this be done without sacrificing enterprise values and expectations and also without going broke in the process or failing to reach your enterprise objectives for the brave new world of smart devices?* Many of the related challenges are shown in Exhibit 1, at the top of the next page.

Looking “Under the Hood” – A Further Examination of the Requirements

At a very high level, “going mobile” with enterprise processes and data seems to be straightforward yet extremely complicated by all of the combinations of possibilities that require development and testing prior to distribution and real-world use. That being said, if you can limit your smart device targets to only one or two (like an iPhone and iPad of a certain generation), your effort may be more manageable. Some enterprises have done this, effectively saying that all of the field representatives will use a specific smart device. This works especially well when the enterprise is providing the target device(s) to the users. However, we seem to be headed into a largely BYOD world, where the decision on which endpoint device is left to the users, in one way or another and often in many ways (such as personalization). The most vivid situation is the one with the most variations – *when the end users are your customers*. **If you want to take on whatever smart device your customer is using, which is a very common situation, then you have to address all of the possible combinations of devices (platform (architecture and operating environment), physical capabilities (like available memory and screen characteristics and accessories, like GPS**

¹⁵ While the nature of the end-users’ experience is set with high expectations, the typical smart device user has grown used to receiving quick and “good-enough” services from the Internet and the Cloud. The key point here is that there is a broad distance between the “near perfection” demanded of most enterprise data center solutions and the “just make it good enough” expectations of many-if-not-most mobile users. What is unacceptable is not delivering something that is “good enough”!

Exhibit 1 — Mobile Brings Business and IT Challenges

- **Bring Your Own Device to Work (“BYOD”)**
 - 200 Million employees do so today
- **Cross-platform Development Considerations**
 - Ability to create the user interface that you need
 - Avoiding the lowest-common-denominator pitfall
 - Learning curve of latest phone features
 - Avoiding vendor lock-in / technology that won’t keep up
- **Using What the Device Has to Offer**
- **R&D Processes and Developer Teams**
- **Back-end Data Integration**
- **Security and Authentication**
- **Post-deployment control of apps**



Source: 2011 IBM Tech Trends Report

Source: IBM

and camera), smart device settings, etc.

This presents many complications and the very real possibility that what needs to be done (and maintained and updated) may actually never be finished¹⁶, which of course is the goal, because time is of the essence and change is continual. **You need to think about what you need in terms of objectives (i.e., the business requirements for data access, sharing and creation/updating) and what you need in terms of breadth of target devices (i.e., the technical requirements for the set of target devices). Then you have to figure how you are going to do this – effectively and efficiently.**

Enterprise (Business) Requirements

Let’s subdivide this into the following activities and then explore each. Be aware that there is overlap here.

- ***Controlled remote access to enterprise data***
- ***Controlled remote access to enterprise data processing***
- ***Inclusion of remote device capabilities in enterprise processes***

Remember that the object of our interest in this discussion is centered on mainframe data and processes. What follows may be applicable more broadly, but the discussion is mainframe-focused. Additionally, “remote access” and “mobile access”

¹⁶ Because the pace of new smart device introductions and/or enhancements to the existing smart device operating environments is very fast, with possibly two generations being introduced in a year (one every six months).

both refer to access by smart devices. Let’s also presume that security of data and processes is a critical requirement.

Controlled Remote Access to Enterprise Data

Data generally will fall into two sub-categories – *historical* (static) and *active* (“in play”, in some way). While your business may see them as an integrated whole, there are important differences.

- **Much of what end users want is access to historical data**, such as what a customer might have ordered previously from an online seller, or a transaction that previously occurred, such as a banking deposit or mutual fund purchase. For data mining, much of what is being mined is historical.
- **Sometimes, what they want is the current (latest) data**, such as the current price and availability of an item (e.g., a specific seat on a specific flight on a specific date) or something based on the current data (like the value of an investment portfolio). In an era of business analytics, real-time data is being analyzed, either as the most current data available or even as it flows in as a real-time data stream.
- **Historical and active data may be part of the same data collection** (e.g., the same database or collection of related databases) or they may be separate. Separate data sources may need to be merged to present what the user is seeking (e.g., when an investor wants to know how his/her current investment portfolio has changed since the first of the year, or since yesterday, when the

market closed). Bring data together from multiple sources compounds the challenges. **What you do not want to do is have to create (without very good reasons) even more databases than you now have, just to satisfy the need to access mainframe-centered data.**

- **The first requirement is for control.** Controlled access, whether remote or internal, requires real-time authentication of the user and policy-based knowledge of permitted access and uses for that user. If your enterprise data is on a mainframe, you already have well-tested processes for processing authentication and access rights. **What you do not want to do is add another (alternative) set of authentication and access management tools and procedures.** Depending on the requirement, if the user cannot be authenticated, you can exclude access or limit access to publicly-available information.

Controlled Remote Access to Enterprise Data Processing

For simplicity, let's call this *transaction processing*, which means "doing business" in some transaction-oriented, data-changing way. Whether reserving and paying for a seat on a train, buying an item from an online retailer, or initiating an online insurance claim, these all are data creating and/or changing activities. This differentiates *data access* from *data processing*. Surfing the web without ordering is without transaction processing and thus only may be data access. Once you place an order or change your personal information (such as a delivery address), you are creating (or modifying) data (hence, the data processing moniker).

On the mainframe, most transaction processing is done via *CICS*. For simplicity, let's assume that you want also want to access data sitting behind *CICS* (most of which is stored in *DB2*, *IMS* and/or *VSAM* databases).

Inclusion of Remote Device Capabilities in Enterprise Processes

So far, the requirements have focused on the data and processes on the mainframe and figuring out how to use a smart device to serve as an active, intelligent endpoint in the relationship. However, smart devices offer some new twists and possibilities. The most interesting possibilities seem to be location-based, which requires an active GPS in the smart device. Most smart devices have this capability. The new possibilities for using the GPS in conjunction with mainframe-based data and processes are endless, but all derive benefit from knowing where the user is located *now*. For example, if I am in a store (or very close to one) and I

want to get a price or details about a product or service, the solution might make a recommendation based on my prior purchases, expressed interests or opinions, or personal profile – all tuned to where I am or even where I am headed. This turns the smart device into a radar of sorts¹⁷, allowing enterprise processes to include location as a meaningful parameter. To a lesser degree (or maybe requiring a greater imagination), the camera and/or microphone on a device also can be sources of input into enterprise processes. For example, if I take a picture, picture recognition software might be able to determine where I am or have been.

Business Requirements are Evolving

All of the requirements above are stated in general terms. They indicate what you (might) want to do if smart devices are endpoints to mainframe data and processes. Clearly, this is a hot topic and of much current interest. **These days, nothing seems hotter than mobile access and this is expected to continue for many years, so there is no way to hide from this new wave of requirements to access cherished and protected mainframe data.**¹⁸ The principal mobile requirements for enterprises are shown above in Exhibit 2, at the top of the next page. What has not yet been discussed is how to do this, effectively and efficiently.

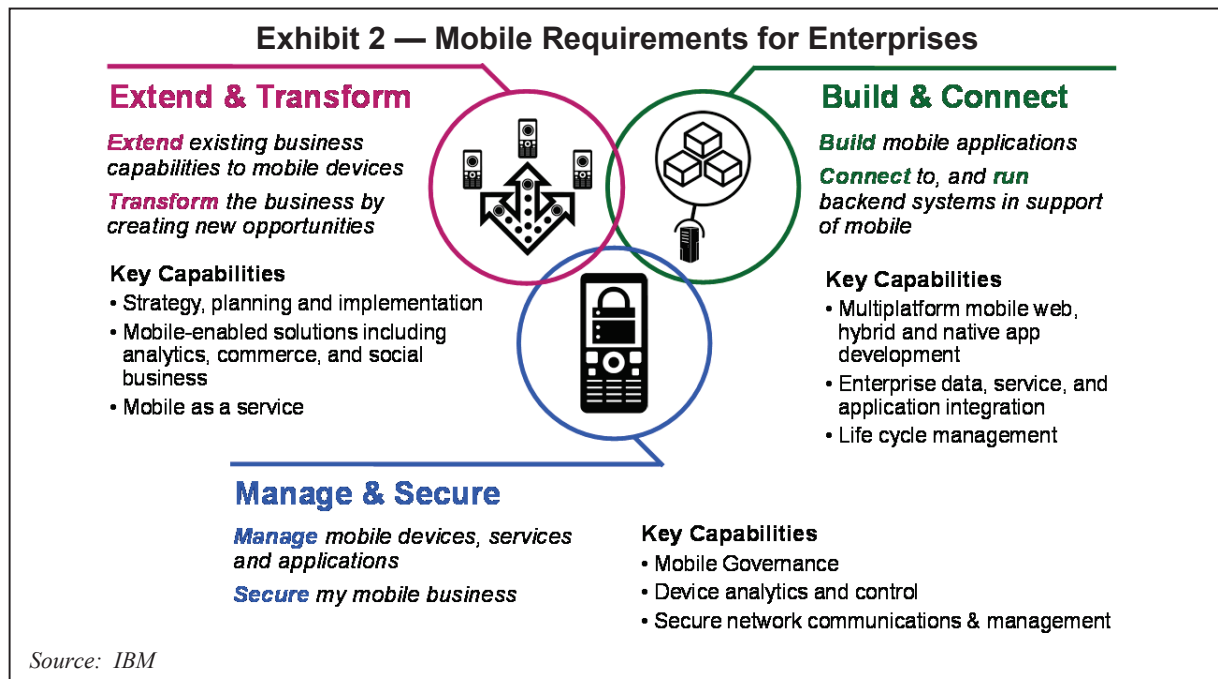
Technical IT Requirements

In the era that preceded the current one focused on mobility and smart devices, we focused on moving from *passive websites* (where information is presented to the user) to *active websites* (where the user could do something with that information, like buy the product just researched or found). Many early attempts at operationalizing websites focused on providing custom-designed and programmed access through the browser. You wrote an application and presented it the user through a very controlled and generically-common portal window (the browser) and you linked that, somehow, to your active, backend processing systems. This was made easier by SOAP-based integrated web solutions (tool kits), such as coding for access via WebSphere Application Server. What was written could be run on all of IBM's server platforms. Thus, there was no longer was a need to develop on or for a specific platform. This made application development and testing simpler and delivery more varied.

For those of you with a large investment in XML and SOAP on the mainframe, your first

¹⁷ If the proper permissions are set to disclose location.

¹⁸ Many consider the growth of cloud computing and big data to be driven by the growth of smart devices.



thought may simply be that you want to now do the same with phones, e.g., use SOA protocols to move data to and from smart device endpoints. This is a reasonable requirement but doing so only delivers a “partial happiness”. It is partial because SOAP presumed a pretty dumb (simple and standard) endpoint – the browser. The browser window on the world (with rich presentation) is ubiquitous, and that is good (very good), but is sort of stuck in a 10-year-ago perspective – let’s call it an “inactively-involved endpoint”. It is inactive in that it does not sense what is happening at the endpoint’s location (in terms of geographic positioning, sights and/or sounds) and then forwards this information for incorporation in enterprise processes. So, while it would be good to be able to tap into the well-understood paths and processes to data via XML and SOAP (i.e., for those deeply-rooted in the traditional world of WebSphere), that is not the only requirement.

Applications in JAVA

What also may be desired or required is a more direct connection from the smart device to those databases and processes on the mainframe, i.e., without knowingly going through SOAP. This also means recognizing that many mobile applications today are written in JAVA or via tools that generate JAVA code or JavaScript. The most common standard is *JSON (JavaScript Object Notation)*¹⁹. JSON is the common language of many

¹⁹ JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a

mobile app programmers. **What you likely want and probably need to do is to bridge the gap between the world of DB2, IMS, VSAM, and CICS²⁰ and the world of JSON – easily.**

Here’s why. Remember those hundreds or thousands of combinations of possible cell phone models, versions, chips, available memory, screen sizes and shapes (and more physical characteristics) and operating systems (and many releases – both standardized and customized by carriers). Well, **you really, really want to write a single mobile application once for all of them and most certainly not a custom one for each** (where each needs to be tested, don’t forget). This is where JSON comes in. **You really want to bridge the gap easily. If you could do this, then you wouldn’t need to keep writing and testing apps developed specifically for a given smart device. JSON is the key to making this happen.**

Digging a Little Deeper – More is Required

Let’s get a little more abstract and then a little more specific. In a BYOD world, the smart phone or tablet user has both personal and enterprise uses. While the best of users might be able to keep these

subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language. See <http://www.json.org/> for more information.

²⁰ More on CICS and direct smart device to CICS connectivity in a later section.

separate (in spirit) or isolated (by design), you can't presume that all of your users are that exceptional or ethically-inclined. **What you need is some way to isolate your enterprise apps and data from whatever else is running on the smart device, including malware.** There are several ways (approaches) to doing this, but all involve some kind of "containerization", basically separating the enterprise environment (apps and data) from that which is something else (personal). A moat, of some kind, usually is placed around the enterprise apps and data, with a prescribed and controlled drawbridge providing the only access path. This can be done *natively* (with separate containers provided by the smart device vendor), by a *third-party application* (which creates the moated environment, potentially with a different look and feel than the native phone experience), or via *virtualization* (where the hypervisor allows multiple environments to run (separately, i.e., isolated) on a given platform).

This all digresses very quickly into an architectural and philosophical discussion of native management versus containerization, a subject worthy of a separate paper, if not several. For now, think about this in terms of what you want – ease of programming and implementation, ability to reach into mainframe data, ability to control, protect, and isolate (if needed) enterprise data and processes. It should be clear where you want and need to go. Finding the right way(s) to do this is the current marathon in which many vendors are running.

Enterprise Messaging

Do consider the importance of enterprise messaging in this big picture. Traditionally, those outside of the data center need to cross the moat and to get to the data inside and access ("pull") the data that they need, typically via HTTP, from a known source. In an era of smart devices, data often is "pushed" out to the remote endpoint, whether by programmed logic or a by triggering event or mobile user request. Now, there is a need for an event-driven paradigm:

- Emitting information from one to many
- Listening for events whenever they happen
- Distributing minimal packets of data in huge volumes
- Pushing information over unreliable networks
- Reliably completing mobile business transactions

Clearly, this is different than handling messaging on desktops and notebooks.

Device and App Management

Here are some important device management questions to consider.

1. ***What is the desired user experience?***
 - Do you want a native app experience versus that of third-party app replications?
 - Do you want a device-wide passcode versus a container-level passcode?
2. ***Do you want to support third-party apps?*** (These are apps provided by someone other than the smart device operating environment vendor.)
 - Do you want to control or manage third-party apps, especially if you are using containers?
 - If your enterprise is using a container approach for its apps and data, how do third-party apps get placed into container?
3. ***Are you concerned about data leakage and controlling it?***
 - Are there deal-breaker data insecurities with native OS capabilities (on each platform)?
4. ***Is privacy and data separation mandated?***
 - Have you considered the privacy, liability, and management control preferences of user base?
5. ***How do you want to approach mobile app management?***
 - ***Native Management*** – This leverages the native OS capabilities for app management.
 - ***SDK Approach*** – App developers (internal and third-party) incorporate management vendor SDKs and libraries to enable integrated app management.
 - ***App Wrapping and Containerization*** – An app-wrapping technology is used to wrap an app binary with a layer of enterprise security and management capabilities.

Endpoint Management Is Key

For desktops, endpoint management tends to be about provisioning operating systems, patching software, and power management. However, PCs and smart devices usually have many of the same management needs:

- Device inventories
- Security policy management
- Application management
- Device configuration
- Encryption management
- Roaming device support
- Integration with enterprise systems
- Delivering a scalable and secure solution
- Be easy to deploy

- Support multiple operating systems
- Be able to consolidate infrastructure

Also, there are other requirements that are just for smart device management. Let's look at several.

- **Self-Service Portal** allows configuration and management of smart devices. This can be important particularly when the user has a new device or when a device is lost or stolen.
- **Device Wipe** is needed when a device is lost and there is a risk of corporate data being compromised. It generally is used to delete all apps and data (both corporate and personal, or only corporate). This can be initiated without involving the user.
- **Jailbreak/Root Detection** is used to detect an underlying condition that the device has been "rooted", i.e., the installed operating environment has been replaced or sufficiently mutated, thus threatening the security or viability of an enterprise app.
- **Location Info**, discussed previously (on Page 6) as a requirement for personalizing the user's experience and as a variable in what is being shown or sent to the user in the enterprise app or in messaging, relies on the GPS in the device to provide geographic coordinates to the on-board app or back to an enterprise server. A good example of this is when the user initiates a transaction at an ATM. The bank could query the location of the account holder's smart device and, if it is not adjacent to the ATM, the bank could take additional steps to confirm the account holder or could deny access. As another example, a location query might provide information on how the smart device is connected, e.g., by 3G/4G or by Wi-Fi. This knowledge might affect how much data or messaging is being sent or received, and when.
- **Enterprise App Store** is an enterprise-controlled portal whose sole purpose is to get the right version of the app installed on the user's smart device. Sometimes the enterprise app store is a private version of the publicly-available app store for a given brand of smart device.²¹ For most environments, the enterprise has a choice – either put your apps in the vendor's store or put them into a private store, where you control who gets to access the private store. There's another factor with some of the vendor's app stores – the vendor may be the gatekeeper to approving the

app and putting it into its store. Typically, this takes extra time.

Consider the potential app store development cycle shown in Exhibit 3, on the next page. Making this happen speedily and with fluidity may be one of the grand challenges faced by enterprise developers. If you have to tailor each app for each of the dozens, hundreds or more of targeted smart devices, there is no way that this could be done speedily. If you couldn't take advantage of public and private app stores to deliver and update the right app to the right smart device and approved user, roll-out would be jerky at best and likely downright problematic for many users and the help desks that support them. Also, if you are using tools from different vendors to make all of this happen, you not only have a training and test problem, but you may also have a huge integration problem, as you try to put all of the pieces together.

Even More to Consider – Tools for Business Analytics and Development

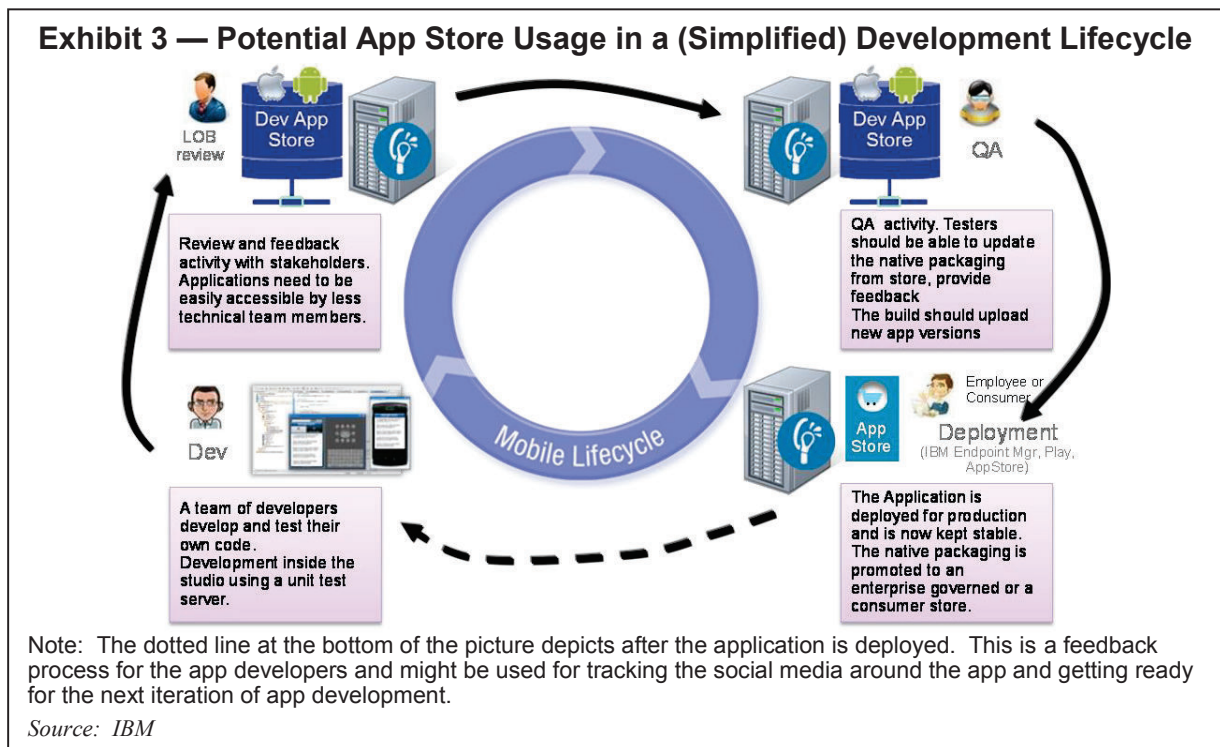
For the mainframe and its mission-critical data, how and where you do enterprise business analytics is a complex subject worthy of another paper, and one was written a few months ago.²² The issue here is that you probably already have tools for business analytics (*Cognos* and *SPSS* from IBM, to mention just two) and tools for development (like the *IBM Rational* toolkit).²³ What you don't want to do is have your solutions for smart devices require you to add to this already long list unnecessarily. Stated differently, you want your smart device solutions (development, test, data management, analytics, implementation and management tool sets) to work together in an integrated, hopefully-familiar way. There are two ways to do this – (a) buy from a single vendor who already has integrated and tested the pieces into a coherent solution set or (b) buy pieces or subsets from multiple vendors and do your own architectural design, integration, and testing. **If time to market is important, doing it yourself should be avoided, especially if it costs no more to procure it all from one vendor. This is especially true now, when the unique challenges of smart devices are the ones that now require the most attention.**

Read on to see how IBM answers all of the above-mentioned requirements.

²¹ Apple's *App Store* and the *Google Play* store, Microsoft's *Windows Store*, and *Samsung Apps* are examples for vendor-supported and controlled app stores.

²² See [The Clipper Group Navigator](#) dated December 12, 2012, entitled *Addressing New Business Analytics Challenges – When the IBM zEnterprise Really Makes Sense*. It is available at <http://www.clipper.com/research/TCG2012030.pdf>.

²³ You might have others or many more.



IBM's Answers

This could be address generically, i.e., without regard for platform, data management middleware, etc., and you might end up at the same place or with the same conclusions. However, the purpose of this paper is to solve the problem of those with data on mainframes, i.e., for those in the data center who want to engage end users with smart devices and, also, for those with business needs. i.e., those in lines of business who want to equip their mobile users with enterprise data access and business transaction processing. More specifically, *how can those who are the keepers of DB2, IMS, and VSAM databases and CICS transaction processes on mainframes make all of this happen, all while garnering support from those in the lines of business?*

IBM MobileFirst Portfolio

IBM's answer is its *MobileFirst* portfolio of products, services, and eco-system of third-party solutions, especially for those whose target includes the mainframe, zEnterprise.²⁴ Given IBM's approach, the same tools can be used to target the mainframe and other server platforms, which is part of the beauty of what IBM has done. The portfolio starts with a foundation stack of tools and appropriately is called the *IBM Mobile Foundation*. Its principal parts are:

- *IBM Worklight*

- *IBM WebSphere Cast Iron*²⁵
- *IBM Endpoint Manager for Mobile Devices*

Each will be discussed below. At this introductory point, it is important to know that **IBM Mobile Foundation offers a rich, workable solution for bringing endpoints (whether smart devices or desktops/laptops) securely into the data rich world of the mainframe.**

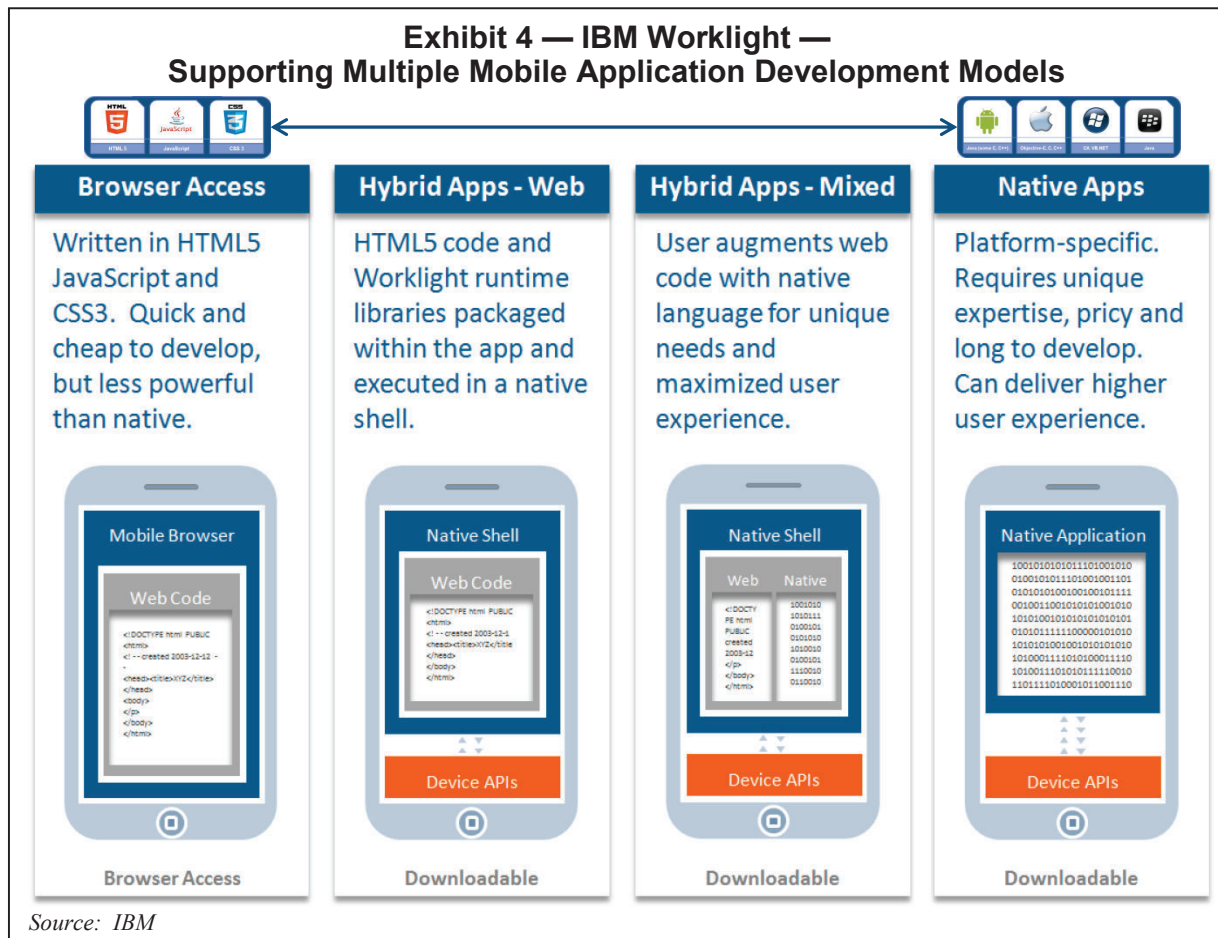
IBM Worklight

Worklight is an application development platform for the mobile enterprise. Its purpose is to speed the development, integration and management of mobile applications by using standards-based technologies and tools to deliver an enterprise-grade services layer. This directly addresses the one-to-very-many problem identified earlier, as Worklight goes beyond mobile app creation to deliver a complete mobile middleware solution. Its key capabilities mobile-optimized middleware include:

- Open approach to 3rd-party integration
- Mix native and HTML
- Strong authentication framework
- Encrypted offline availability
- Enterprise back-end connectivity
- Unified push notifications

²⁵ Cast Iron provides a graphical configuration interface approach – rather than custom coding, on-demand tooling or traditional middleware – to help integrate applications quickly and simply.

²⁴ For the details, see www.ibm.com/mobilefirst.



- Data collection for analytics
- Direct updates and remote disablement
- Packaged runtime skins

These are a good checklist for what you will want in your application development platform. These are the “what”, as in “what you need to do.”

Equally important are the “where” and “how” all of this will happen, as described in Exhibit 4, above. As discussed earlier, there is no “one right way” to approach application development. In fact, in the largeness of an enterprise, it may be right to use many different approaches. **What you want is a development vehicle that does not limit you to doing it in just one way.** Worklight provides this range of approaches within the same framework.

Worklight’s Major Components

There are four major Worklight components, as shown in Exhibit 5, at the top of the next page.

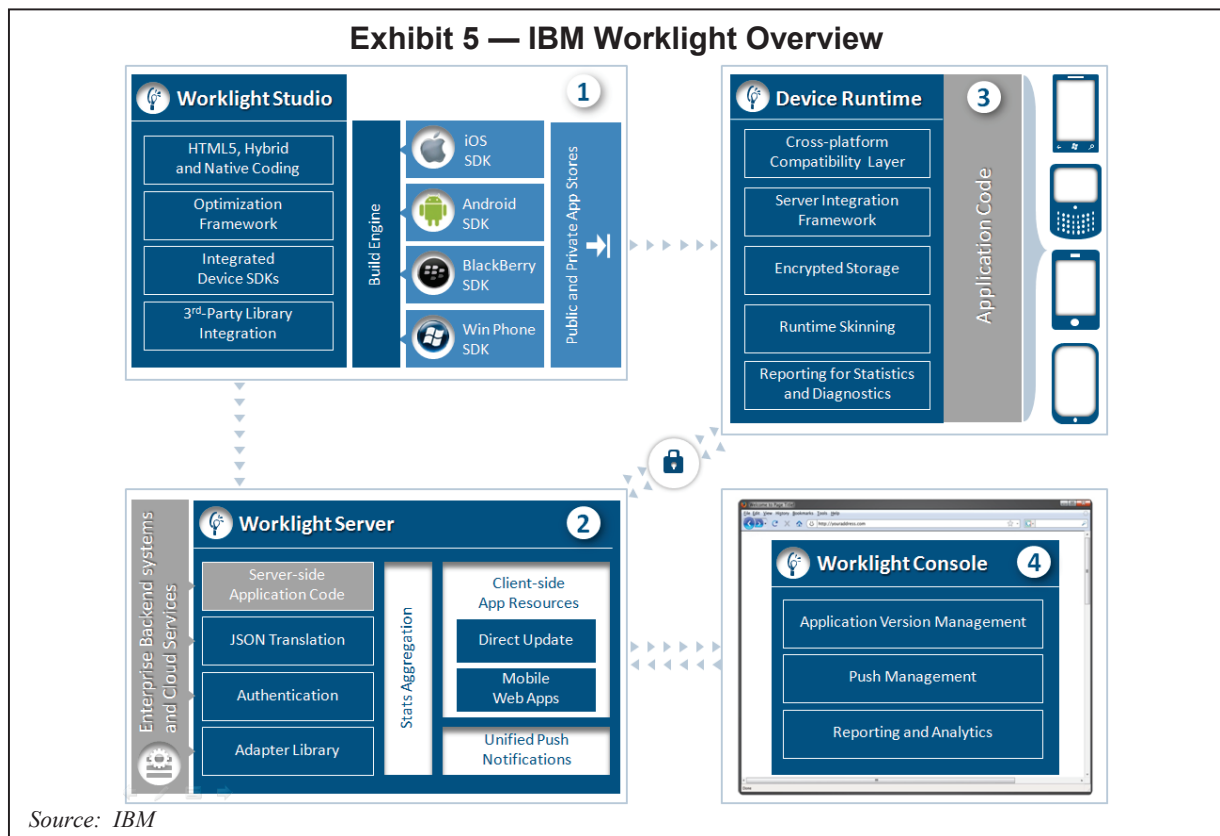
- **IBM Worklight Studio** is an Eclipse-based IDE, allowing developers to perform all the coding and integration tasks required to develop a fully operational mobile application. (See Exhibit 6, at the bottom of the next page.)

- **IBM Worklight Server** is a JAVA-based server that is a scalable gateway between applications, external services, and an enterprise’s backend infrastructure. It performs data translation to streamline backend data for smart device consumption, supports physical clustering for high availability, manages push notifications, controls application deployment and versioning, and provides analytics on user adoption and usage. It contains security features to enable connectivity, multi-source data extraction and manipulation, authentication, direct update of web and hybrid apps, analytics and operational management functions.

- **IBM Worklight Device Runtime Components** are client-side runtime code that embeds server functionality within the target-environment of deployed apps.

- **IBM Worklight Console** is a web-based UI dedicated for the ongoing monitoring and administration of the Worklight Server and its deployed apps, adapters, and push notifications.

There are some additional capabilities, which represent some very important functionality.



IBM Worklight Application Center

This is an enterprise app store that helps organizations govern the distribution of production-ready mobile apps across the enterprise and also to elicit and organize user feedback. The Application Center also can be used by development teams to automate the distribution of pre-release software versions and analyze feedback by version and device,

accelerating cycle time.

Worklight Supports Multiple Mainframe Environments

zEnterprise offers many operating environments, most notably z/OS and *Linux for System z*. Currently, Worklight supports a variety of Linux releases and some popular middleware releases (*WebSphere Application Server Network Deploy-*

Exhibit 6 — IBM Worklight Studio

Rich page editor

- Worklight studio optimized for development of hybrid mobile applications, including a rich -page, drag-and-drop editor, palette of components, properties view, beautified outline, and raw editable source code for immediate UI customization.

Enhanced desktop preview

- A new browser simulator allows the definition of the form factor of the target mobile device, concurrently displaying multiple devices on the screen, and simulating Cordova APIs.

Apache Cordova is a platform for building native mobile applications using HTML, CSS (Cascading Style Sheets), and JavaScript

Source: IBM

Exhibit 7 — IBM Endpoint Manager Elements



Single intelligent agent

- On workstations, servers and mobile devices
- Continuous self-assessment
- Continuous policy enforcement
- Minimal system impact (<2% CPU, <10MB RAM)



Flexible policy language (Fixlets)

- Thousands of out-of-the-box policies
- Best practices for operations and security
- Simple custom policy authoring
- Highly extensible/applicable across all platforms



Single server and console

- Highly secure, highly available
- Aggregates data, analyses and reports
- Manages up to 250K endpoints per server



Virtual infrastructure

- Designate Endpoint Manager agents as a relay or discovery point in minutes
- Provides built-in redundancy
- Leverages existing systems/shared infrastructure

Source: IBM

ment V7.0 or higher and future releases, modification levels, and fix packs and *Apache Tomcat 7.0* and future fix packs).

IBM Worklight Server Connectivity

IBM Worklight Server connects with other middleware via adapters, including ones for SAP, SOAP, REST, SQL, and JMS. It integrates with software as a service (SaaS) cloud products via *IBM WebSphere Cast Iron* and enables enterprises to integrate cloud and on-premise applications in days, reduce integration costs and optimize resources and productivity in SaaS and cloud models. Worklight performs data transformation to streamline backend data for mobile consumption and it provides server and device security control. It supports physical clustering for high availability and controls application deployment and versioning. Worklight uses push notification for administration and provides adoption and usage analytics.

Worklight provide a graphical configuration interface approach – rather than custom coding, on-demand tooling or traditional middleware – to help integrate applications quickly and simply. They use pre-configured templates based on common integration scenarios to accelerate integration. In addition, these manage server and device security, performs data translation to streamline backend data for smart device consumption, supports physical clustering for high availability, manages push notifications, controls application deployment and versioning, and provides analytics on user adoption

and usage.

IBM Endpoint Manager for Mobile Devices

It should not be a surprise that IBM has an answer (maybe better described as a collection of answers) on how to manage all of these endpoint smart devices. *IBM Endpoint Manager for Mobile Devices* provides a completely integrated approach for managing, securing, and reporting on laptops, desktops, servers, smartphones, tablets, and even specialty devices, such as point-of-sale terminals. This provides customers with real-time visibility and control over all devices that employees and others use in their daily job functions; reducing costs, increasing productivity, and improving compliance. It addresses the issues of security, complexity and BYOD policies that challenge support for an increasingly mobile workforce.

- **Safeguards enterprise data** with advanced security and compliance features.
- **Delivers enterprise visibility** with a consolidated inventory of employee- and corporate-owned devices.
- **Manages as a unified infrastructure** using a single platform to manage all your enterprise devices.

The Endpoint Manager architecture deals with, connects, and integrates the many pieces of the mobile device puzzle. If you have a much simpler architecture in mind, you are probably excluding some necessary pieces. Endpoint Manager's criti-

Exhibit 8 — New CICS Transaction Server 5.1

Operational Efficiency

- Greater capacity - achieve cost savings through consolidation
- Managed operations - reduce cost and risk through automation
- Increased availability - reduce the need for planned downtime
- Deeper insight - improve decision making and audit readiness

Service Agility

- First-class applications - create agile services from existing assets
- First-class platforms - create agile service delivery platforms
- Modern interfaces - build rich web experiences for critical applications
- Foundational enhancements - extend core capabilities

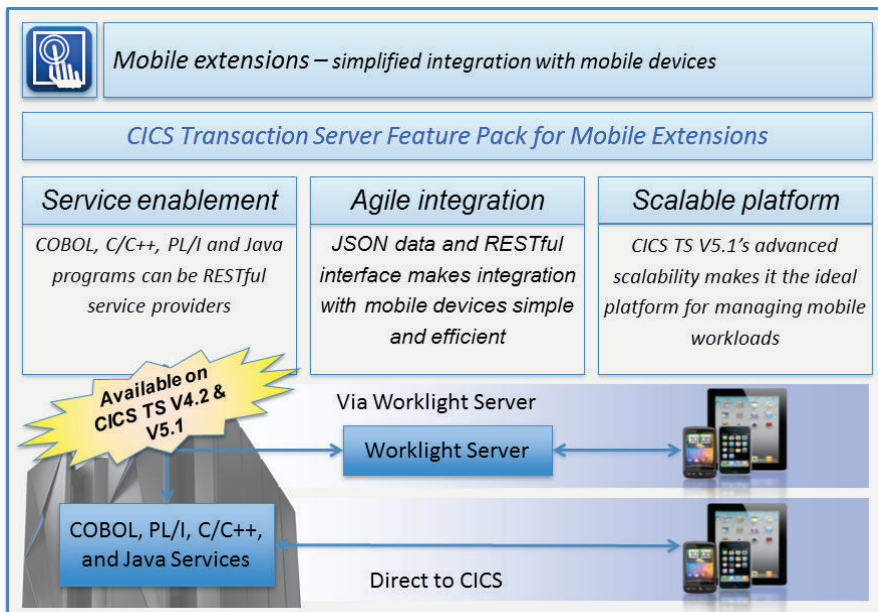
125+ requirements satisfied!

... with Cloud Enablement

consistent with the IBM Cloud Computing strategy
positions customers for the next transformational era in technology
moves towards a cloud oriented service delivery platform

Source: IBM

Exhibit 9 — CICS Transaction Server Feature Pack for Mobile Extensions



Source: IBM

cal elements are shown in Exhibit 7, on the previous page.

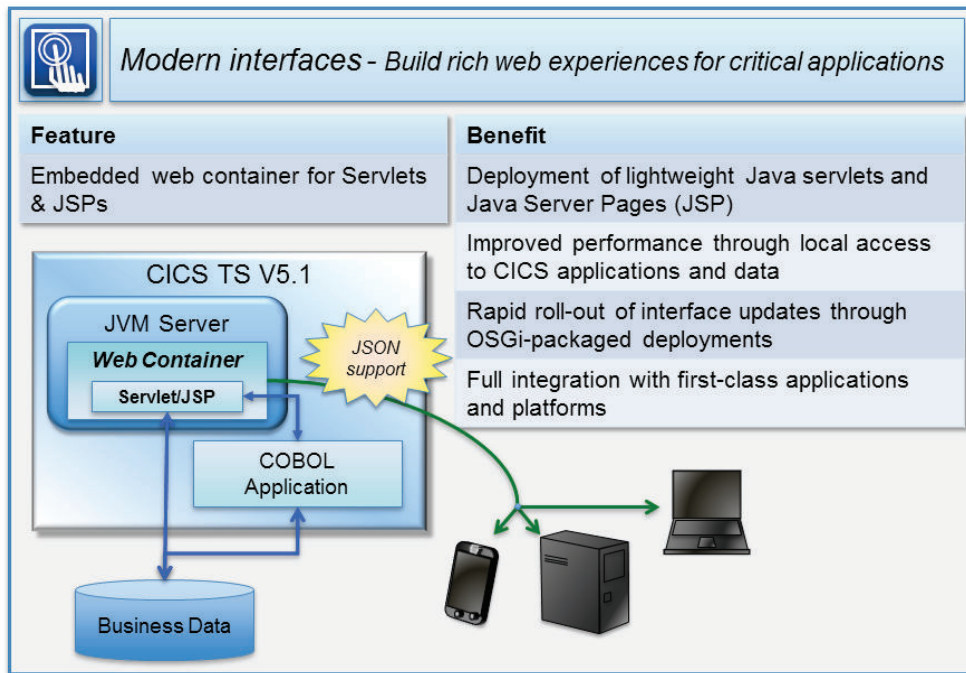
IBM CICS

V5.1 is a new version of CICS Transaction Server (CICS TS). While that is not required for interaction with smart devices, it is a detail that might affect what you are doing, in a very positive way. See Exhibit 8, above, for an overview.

CICS Transaction Server Feature Pack for Mobile Extensions

CICS Transaction Server Feature Pack for Mobile Extensions is directly related to integration of CICS and smart devices, as shown in Exhibit 9, also above. This targets Transaction Server V4.2 and V5.1 and provides two ways to get from smart devices to CICS (and vice versa). The first is through Worklight Server and the second is a direct

Exhibit 10 — CICS Interfaces



Source: IBM

connection into CICS from COBOL, PL/1, C/C++, and Java Services. These smart device extensions simplify integration and access. It is important to note that you do not have to be on the latest version of CICS (5.1) to use these extensions, which means that if you are on V4.2, you do not have wait for V5.1 to be procured, installed, and applications updated and tested. Thus, you can begin mobile access to mainframe data much sooner.

IBM just announced some new CICS capabilities related to mobile applications. First, IBM is delivering JSON support to the CICS transaction server running on z/OS. That is shown in yellow on the diagram on Exhibit 10, above. Second, IBM now is offering a lower-cost licensing charge for qualifying new CICS Transaction Server workloads on zNALC LPARs. In addition, qualified new CICS TS applications, including approved

Exhibit 11 — Rational IDEs with IBM Worklight

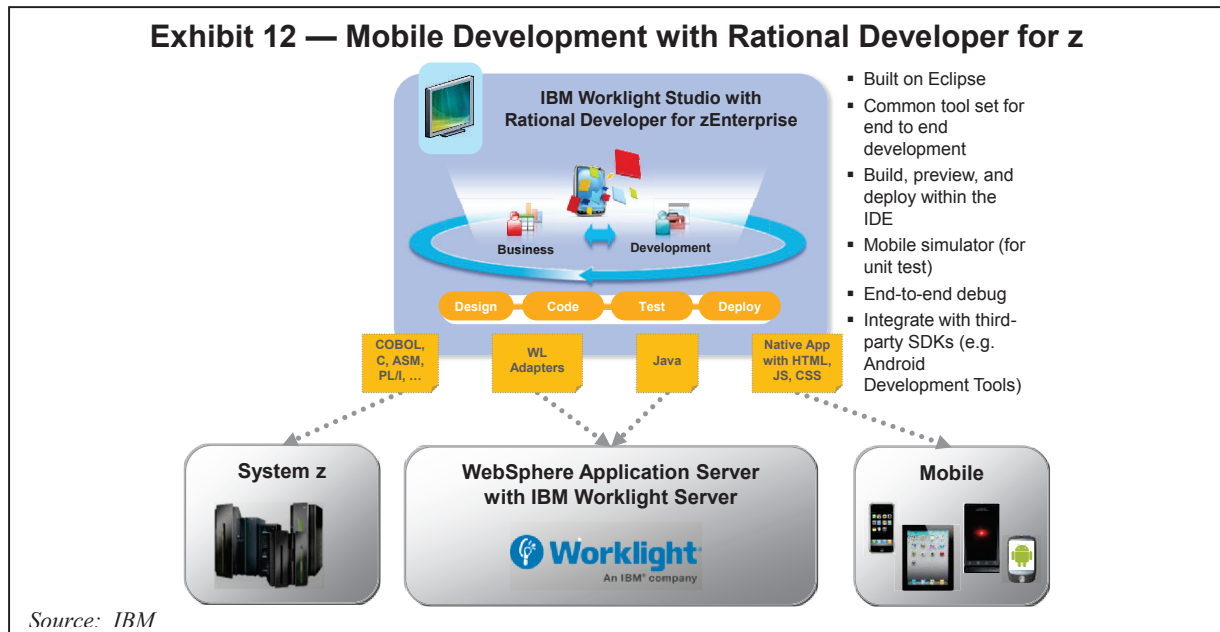
Design, code, build, test, and deploy mobile apps that run on a wide variety of mobile platforms; extend existing back-end services and data to mobile apps



IBM Worklight is included in the following IDEs (for development purposes only):

- Rational Developer for zEnterprise v8.5
- Rational Developer for Power Systems v8.5
- Rational Application Developer v8.5
- Rational Software Architect v8.5

Source: IBM



mobile and service-enabled applications running in the CICS TS Java Virtual Machine (JVM) Server, will be eligible for CICS TS one-time-charge (OTC) pricing. What this all means is that if you connect smart devices with the CICS Transaction Server, you may be eligible for significant cost savings by running in a zNALC partition. **In effect, IBM is encouraging you to put your smart device new workloads on zEnterprise by giving you a hefty financial incentive.**

IBM Rational Developer for z

But there is more! If IBM Rational Developer is part of your enterprise IT toolbox, then you have one more way to construct, debug, and test mobile user interfaces as part of your efforts to access data and processes on zEnterprise from smart devices. (See Exhibit 11, at the bottom of the previous page.) This is the way to tie Rational IDE tools for zEnterprise into the big picture, as shown in Exhibit 12, above. This is how it all comes together.

Conclusion

A lot of territory has been covered in this paper, from the challenges of many thousands of smart devices to getting at mainframe data and processes from these devices to doing this in an enterprise-class way and finding efficiencies along the way. As described, there are many ways to do this. What you decide to do and how you will do it most likely will largely depend on what software (middleware and tools) you are using today on zEnterprise. Your goal should be not only to make this all happen, but to do it efficiently and effec-

tively. By definition, if you have zEnterprise as your mission-critical server, you already are looking to IBM for answers, approaches, and architectures. Now the ball is in your court. From your perspective in enterprise IT, you now need to take a closer look at how well IBM's pieces fit together into a workable and effective solution. Don't delay; the clock already is ticking!



About The Clipper Group, Inc.

The Clipper Group, Inc., now in its twenty-first year, is an independent publishing and consulting firm specializing in acquisition decisions and strategic advice regarding complex, enterprise-class information technologies. Our team of industry professionals averages more than 25 years of real-world experience. A team of staff consultants augments our capabilities, with significant experience across a broad spectrum of applications and environments.

➤ *The Clipper Group can be reached at 781-235-0085 and found on the web at www.clipper.com.*

About the Author

Mike Kahn is Managing Director and a co-founder of The Clipper Group. Mr. Kahn is a veteran of the computer industry, having spent more than four decades working on information technology, spending the last 20 years at Clipper. For the vendor community, Mr. Kahn specializes on strategic marketing issues, especially for new and costly technologies and services, competitive analysis, and sales support. For the end-user community, he focuses on mission-critical information management decisions. Prior positions held by Mr. Kahn include: at International Data Corporation – Director of the Competitive Resource Center, Director of Consulting for the Software Research Group, and Director of the Systems Integration Program; at Power Factor Corporation, a Boston-based electronics start-up – President; at Honeywell Bull – Director of International Marketing and Support; at Honeywell Information Systems – Director of Marketing and Director of Strategy, Technology and Research; at Arthur D. Little, Inc. – a consultant specializing in database management systems and information resource management; and at Intel Corporation – Mr. Kahn served in a variety of field and home office marketing management positions. Earlier, he founded and managed PRISM Associates of Ann Arbor, Michigan, a systems consulting firm specializing in data management products and applications. Mr. Kahn also managed a relational DBMS development group at The University of Michigan, where he earned B.S.E. and M.S.E. degrees in industrial engineering.

➤ *Reach Mike Kahn via e-mail at Mike.Kahn@clipper.com or via phone at (781) 235-0085 Ext. 121. (Please dial "121" when you hear the automated attendant.)*

Regarding Trademarks and Service Marks

The Clipper Group Navigator, The Clipper Group Explorer, The Clipper Group Observer, The Clipper Group Captain's Log, The Clipper Group Voyager, Clipper Notes, and "*clipper.com*" are trademarks of The Clipper Group, Inc., and the clipper ship drawings, "*Navigating Information Technology Horizons*", and "*teraproductivity*" are service marks of The Clipper Group, Inc. The Clipper Group, Inc., reserves all rights regarding its trademarks and service marks. All other trademarks, etc., belong to their respective owners.

Disclosures

Officers and/or employees of The Clipper Group may own as individuals, directly or indirectly, shares in one or more companies discussed in this bulletin. Company policy prohibits any officer or employee from holding more than one percent of the outstanding shares of any company covered by The Clipper Group. The Clipper Group, Inc., has no such equity holdings.

After publication of a bulletin on *clipper.com*, The Clipper Group offers all vendors and users the opportunity to license its publications for a fee, since linking to Clipper's web pages, posting of Clipper documents on other's websites, and printing of hard-copy reprints is not allowed without payment of related fee(s). Less than half of our publications are licensed in this way. In addition, analysts regularly receive briefings from many vendors. Occasionally, Clipper analysts' travel and/or lodging expenses and/or conference fees have been subsidized by a vendor, in order to participate in briefings. The Clipper Group does not charge any professional fees to participate in these information-gathering events. In addition, some vendors sometime provide binders, USB drives containing presentations, and other conference-related paraphernalia to Clipper's analysts.

Regarding the Information in this Issue

The Clipper Group believes the information included in this report to be accurate. Data has been received from a variety of sources, which we believe to be reliable, including manufacturers, distributors, or users of the products discussed herein. The Clipper Group, Inc., cannot be held responsible for any consequential damages resulting from the application of information or opinions contained in this report.