Mobile Cross Platform Development – *really*?

Jonathan Marshall, IBM Mobile Technical Specialist

Objectives

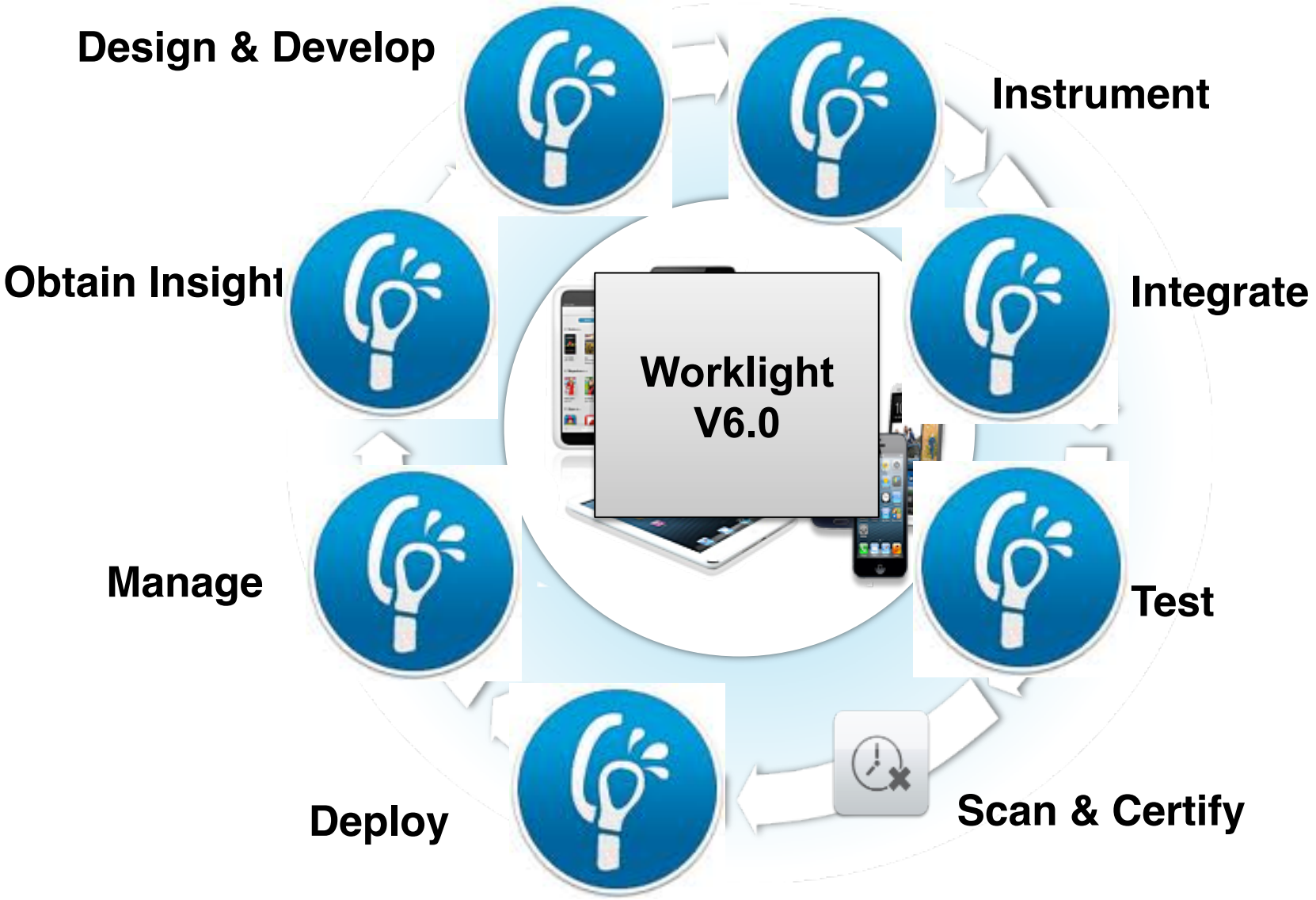
- Worklight update
- Brief demonstration
- Experiences around cross-platform development

IBM MobileFirst Offering Portfolio

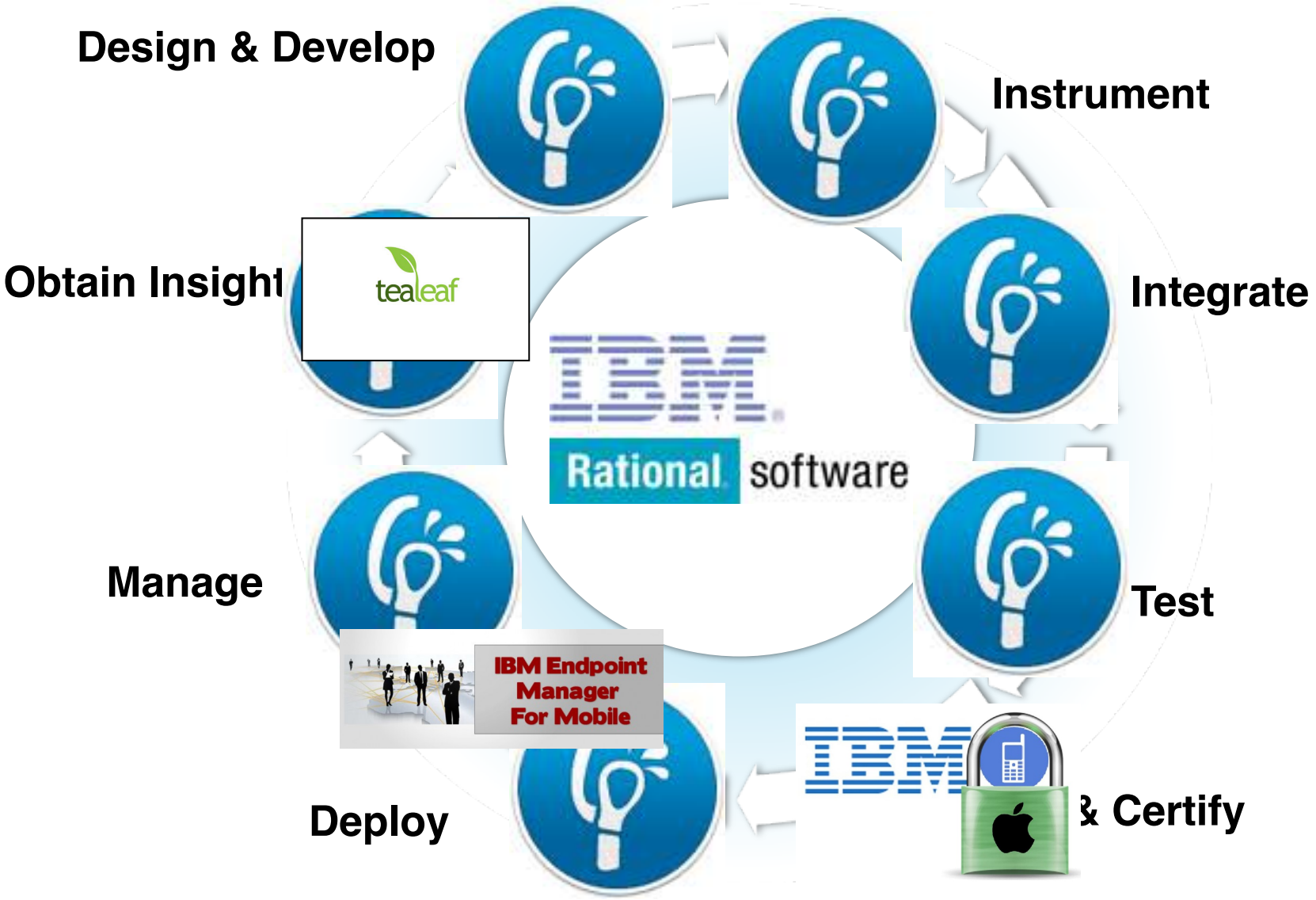
The IBM Mobile Lifecycle

The IBM Mobile Lifecycle

The IBM Mobile Lifecycle

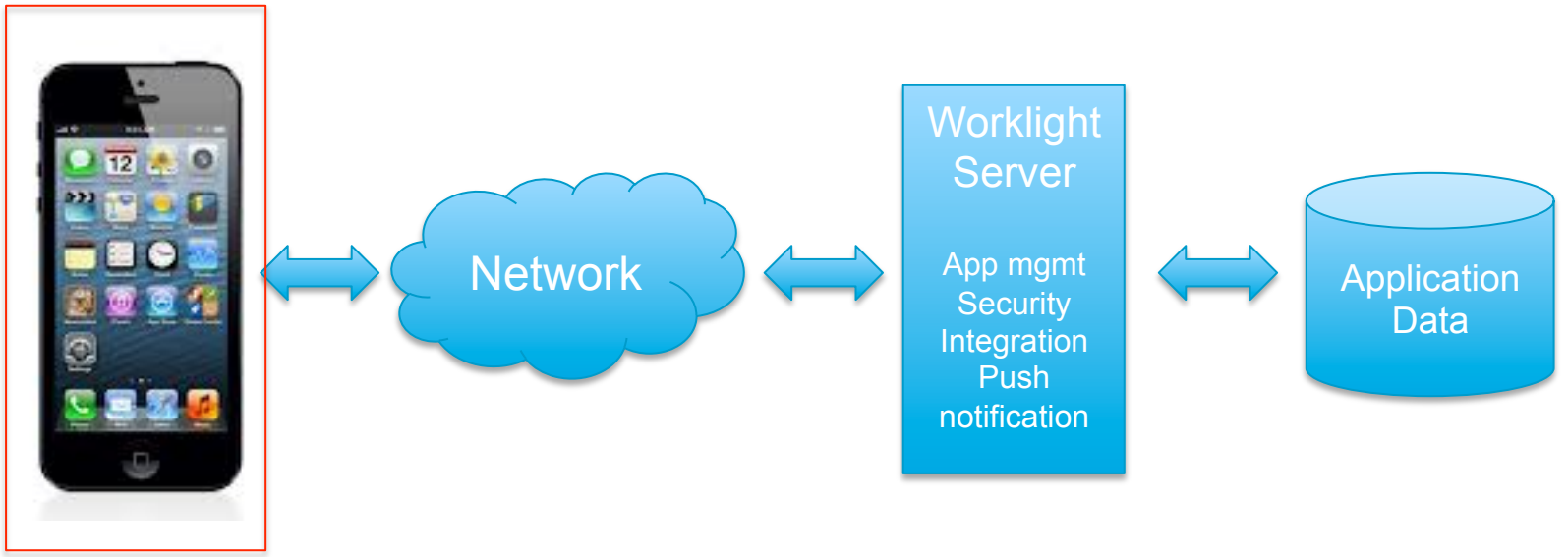


The IBM Mobile Lifecycle

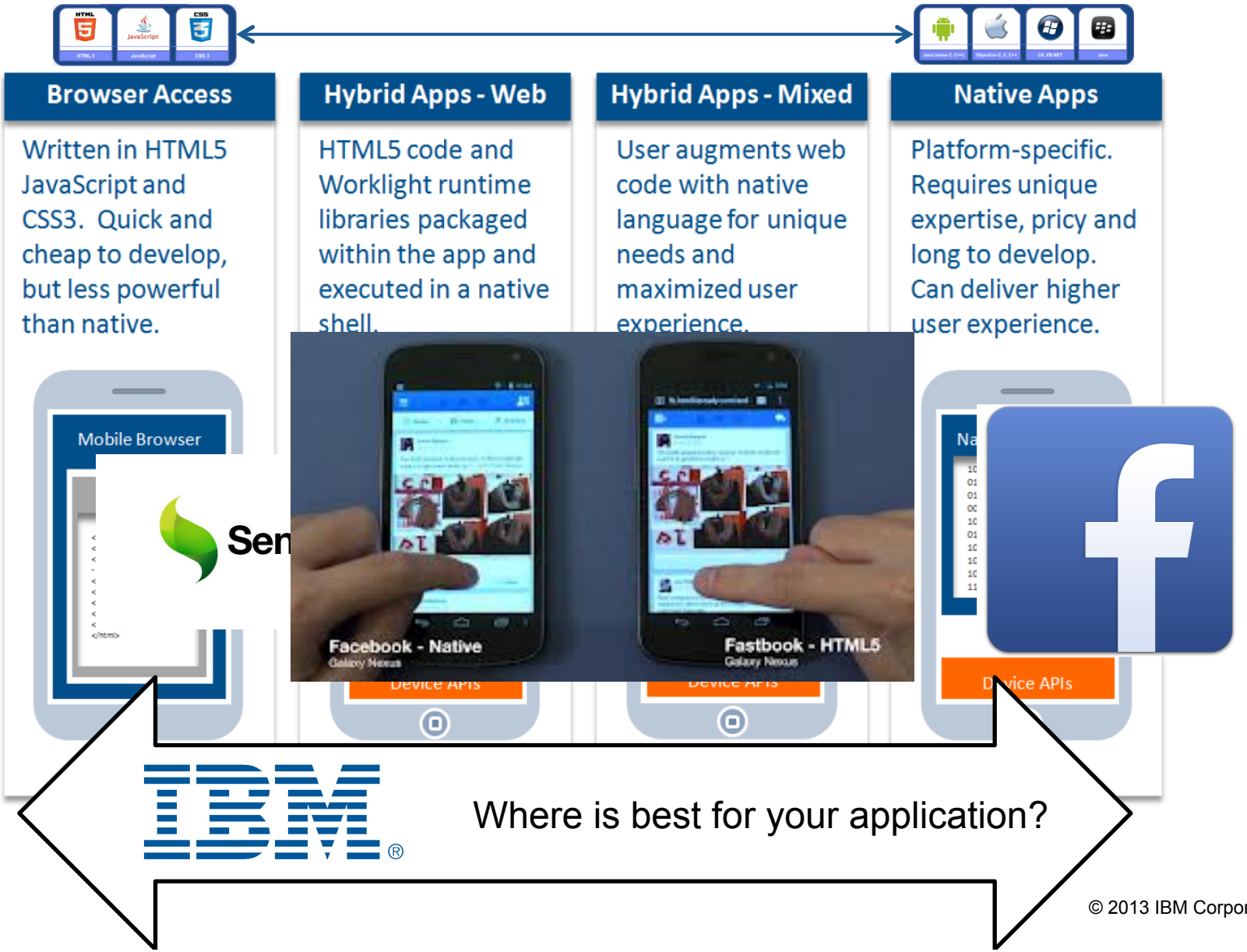


DEMO

Cross platform affects end to end mobile development



Worklight supports multiple mobile development models



What is hybrid? What approaches are there to hybrid?

- Hybrid: “A thing made combining two different elements”

- Development: Cross-platform (productivity and flexibility)
- Deployment: Native (full integration, high user experience)

- Approaches to hybrid

X-platform compilation	Pure HTML5	HTML5 + Native
First generation X-platform Pros Rapid prototyping Great user experience Cons Restricted to widget set provided Platform dev scale	Code HTML5, Deploy Native Pros Openness, Skills Develop once, deploy many Still have device integration Cons “Last 5%” UX	Best of both worlds Code base is reusable Can access any native functionality desired (lose X-platform for that feature)

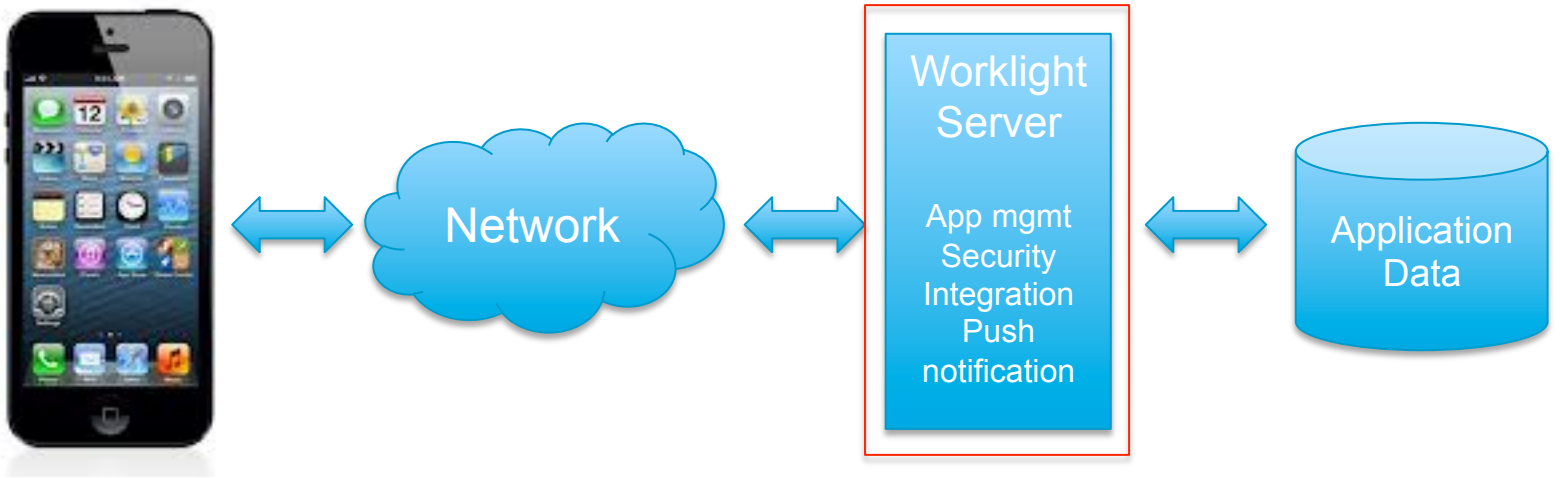
How far can you go with HTML5?

- What's possible?
  - Default to all in HTML5
  - Can emulate native look and feel (or not)
- Device access available in HTML5?
  - Standard function; camera, GPS, etc
- So do we need native?
  - Specific function E.g. QR Code reader
  - Some have chosen specific UX components for native dev, such as slide-in menu
- Device power and capability vary
  - Android hardware acceleration for graphics has been behind that of iOS. Catching up
- Skills are important
  - Basics can rely on web development skill
  - But for high end user experience do need good JavaScript and CSS3 skills
  - And good software engineering practices
  - Rely on a good frameworks such as Dojo, JQuery, Sencha

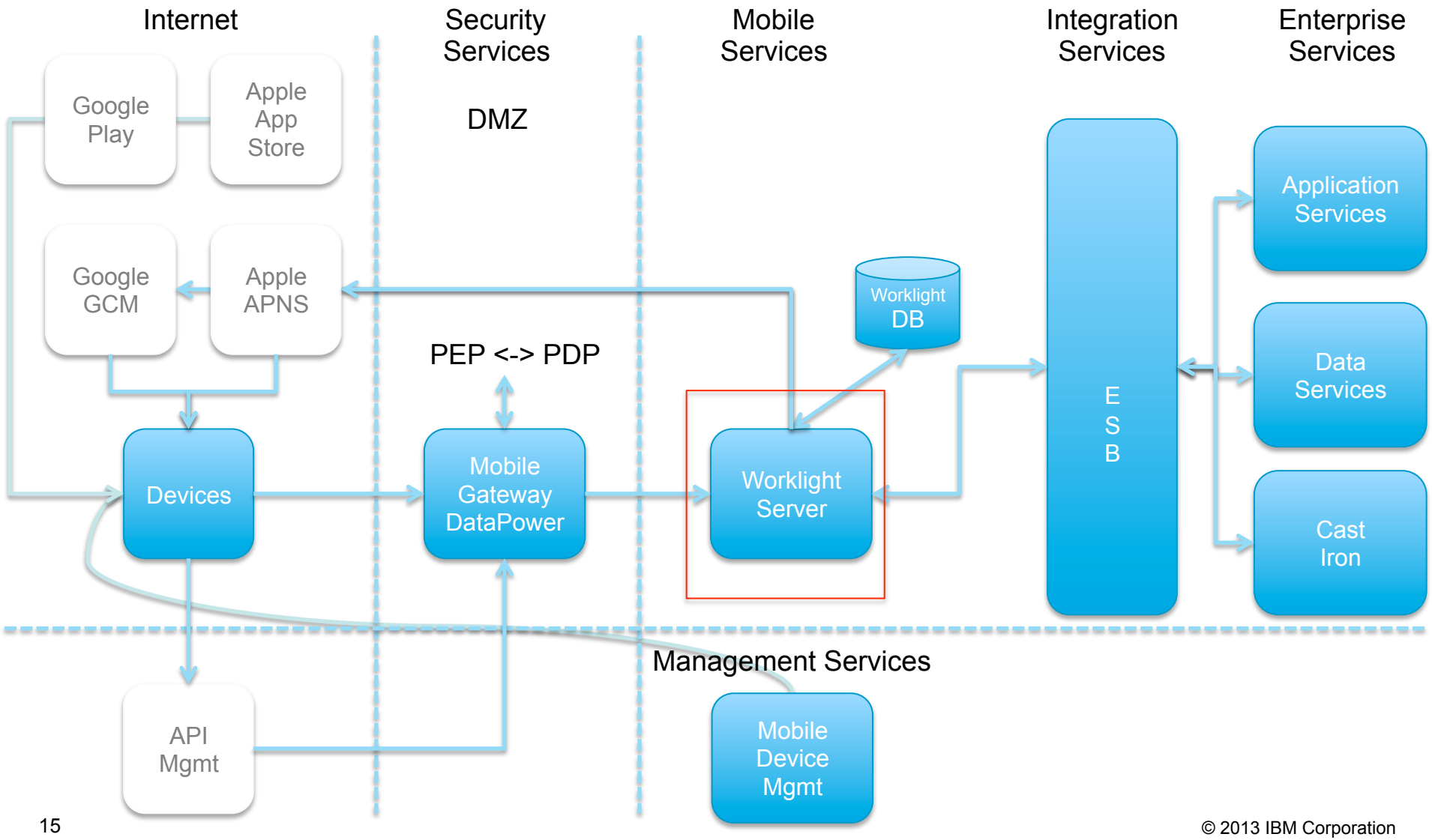
How does Worklight help with cross-platform mobile applications?

- Development Lifecycle
  - Cross-platform dev framework
  - Dev build system
  - Leave OS support to the platform
  - Application Center
  - Testing
- Higher level APIs
  - Apache Cordova – APIs for device access, Designed to be extensible
  - Security
  - Native database with synchronisable storage (JSONStore)
  - Geo fencing
  - “Crash” analytics
- Server-side infrastructure… next few slides

Cross platform affects end to end mobile development



Mobile Infrastructure – The case for mobile middleware



Mobile Infrastructure – The case for mobile middleware

- Mobile-specific
  - Separate from enterprise services
  - Optimise data for the wire – don't need all of it, don't want it in verbose formats (XML)

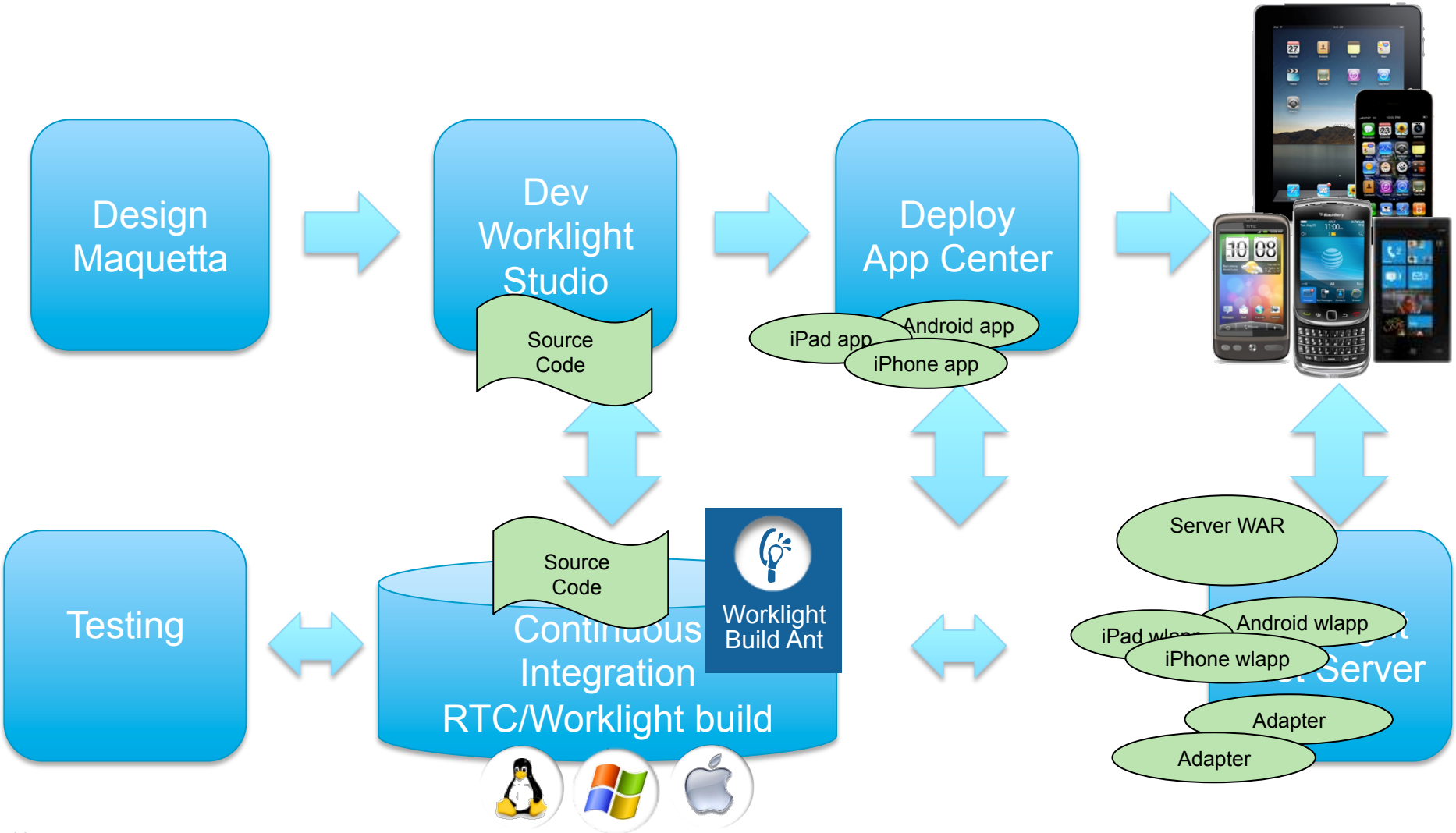
- Application-specific
  - Offload application processing from devices
  - "Mash up" enterprise services – further optimise bandwidth and latency
  - Develop in mobile application language – JavaScript
  - Security for device, application, and users of specific application

- Device-agnostic – *Do it once!*
  - Integration
  - Push notification
  - Management
  - Security

The development lifecycle is getting faster and faster

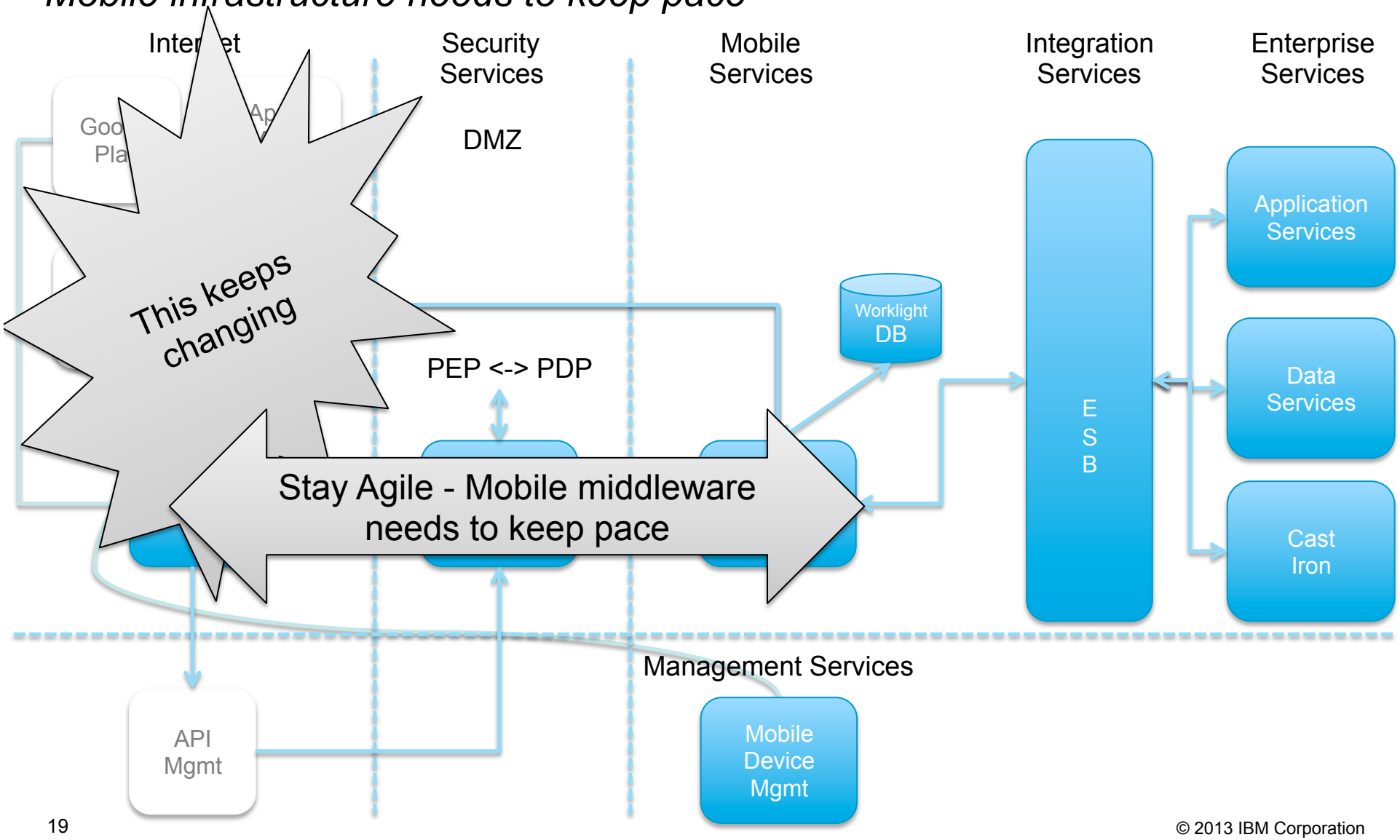
Worklight enables Speed and Agility

Base development on continuous integration

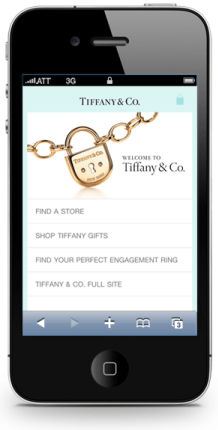


Worklight enables Speed and Agility

Mobile infrastructure needs to keep pace



Summary – Is cross-platform really possible?



Three Ways to Get Started with IBM

1. **Download the free IBM Worklight Developer Edition: ibm.com/developerworks/mobile/worklight**
2. **Download the free IBM Mobile Development Lifecycle Solution (Worklight PLUS ALM): ibm.com/developerworks/cloud/cloudtrial**
   Promo code IMDLS
3. **Learn more: ibm.com/mobilefirst**