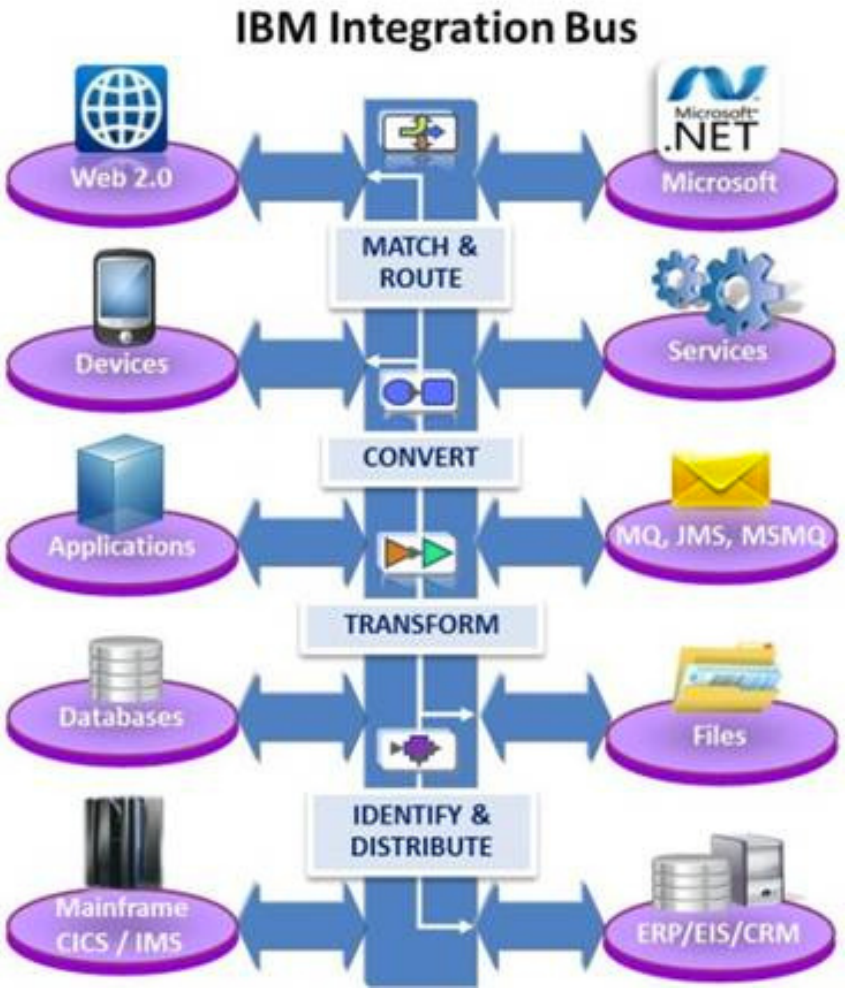
Integrating Mobile apps with your Enterprise

Jonathan Marshall
marshalj@uk.ibm.com

Agenda

- Mobile apps and the enterprise
- Integrating mobile apps with Enterprise Applications
- Mobile apps and IBM Messaging
- Summary

Who's in the audience?



Why enterprises are putting Mobile First

1b Smartphone users by 2016

91% always keep mobile at arms reach

$534b Mobile transactions by 2015

95% of Mobile traffic is data by 2015

75% of users act on location-based offers

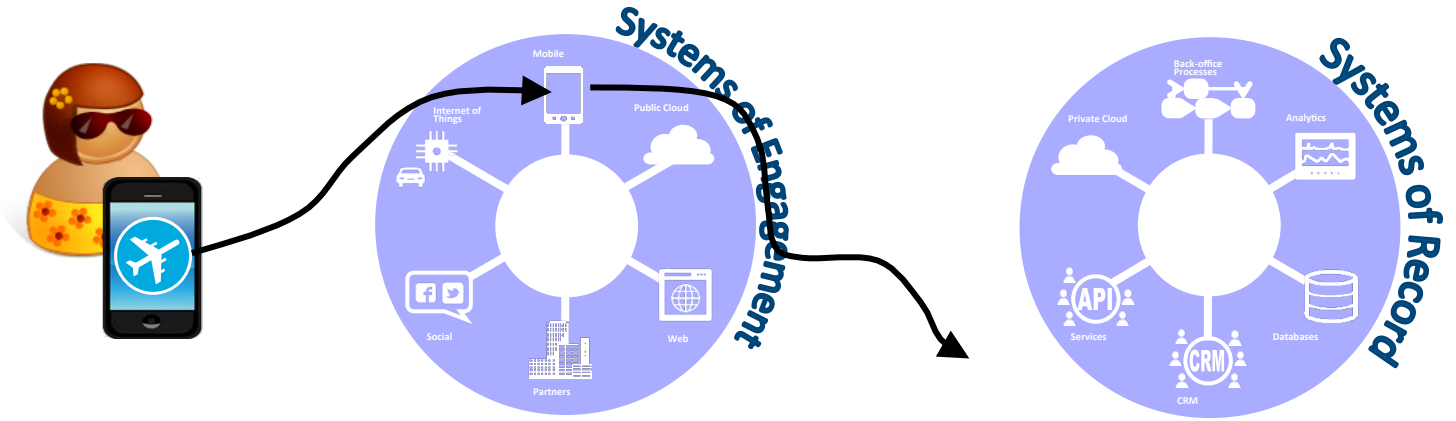
11,000 APIs on the Programmable Web

90% using combination of multiple devices

Or just consider the 2 recent Papal inaugurations…

Un-integrated Mobile apps are not Engaging

Example: Customer uses Mobile app to interact with Airline

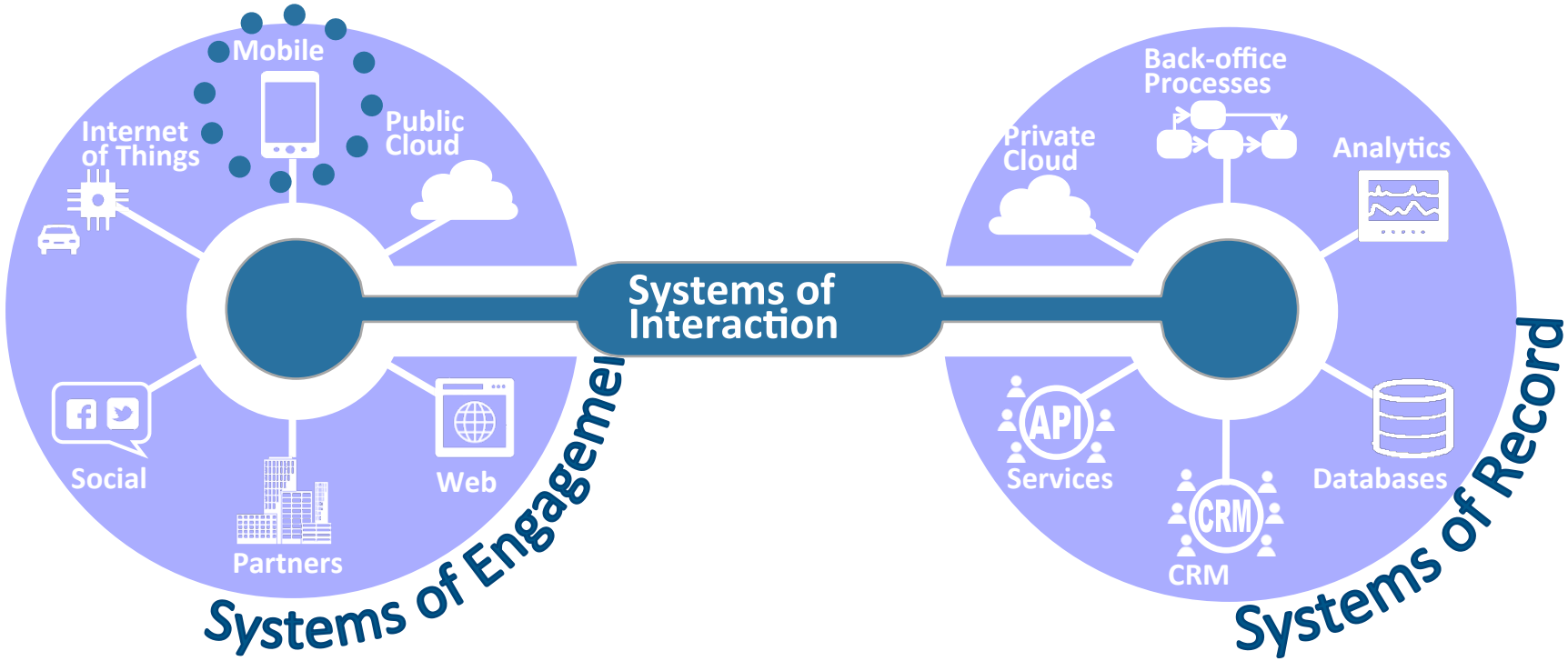


- ✓ **Can** see my ticket
- ✓ **Can** find my seat
- ✓ **Can** see upgrade option
- ✓ **Can** check flight status
- ✓ **Can** see in-flight meal

- ✗ **Can't** change my ticket
- ✗ **Can't** move my seat
- ✗ **Can't** buy upgrade
- ✗ **Can't** change flight
- ✗ **Can't** buy my choice of food

Systems of Interaction

- Able to **innovate rapidly** by combining new and existing capabilities
- Can **pace the rate of change** between future and prior investments
- Interacts **as one** with its clients across a diverse range of touch-points



Challenges to delivering an engaging Mobile experience

Range of Devices

How to cost-effectively support range of popular device platforms?

Pace of Innovation

How to restless reinvent and enhance Mobile experience faster?

Protect Perimeter

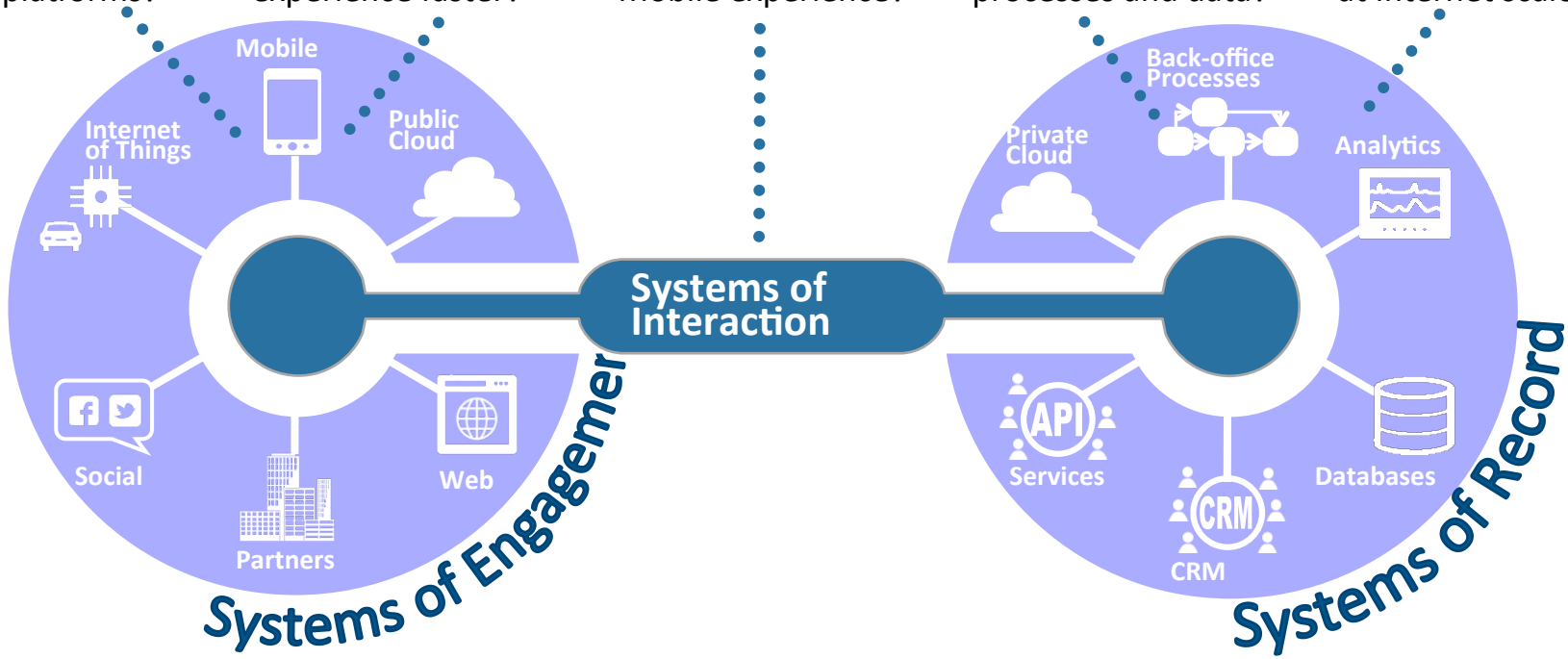
How to secure the boundary without disrupting the Mobile experience?

Backend Integration

How to integrate Mobile activities into existing back-end processes and data?

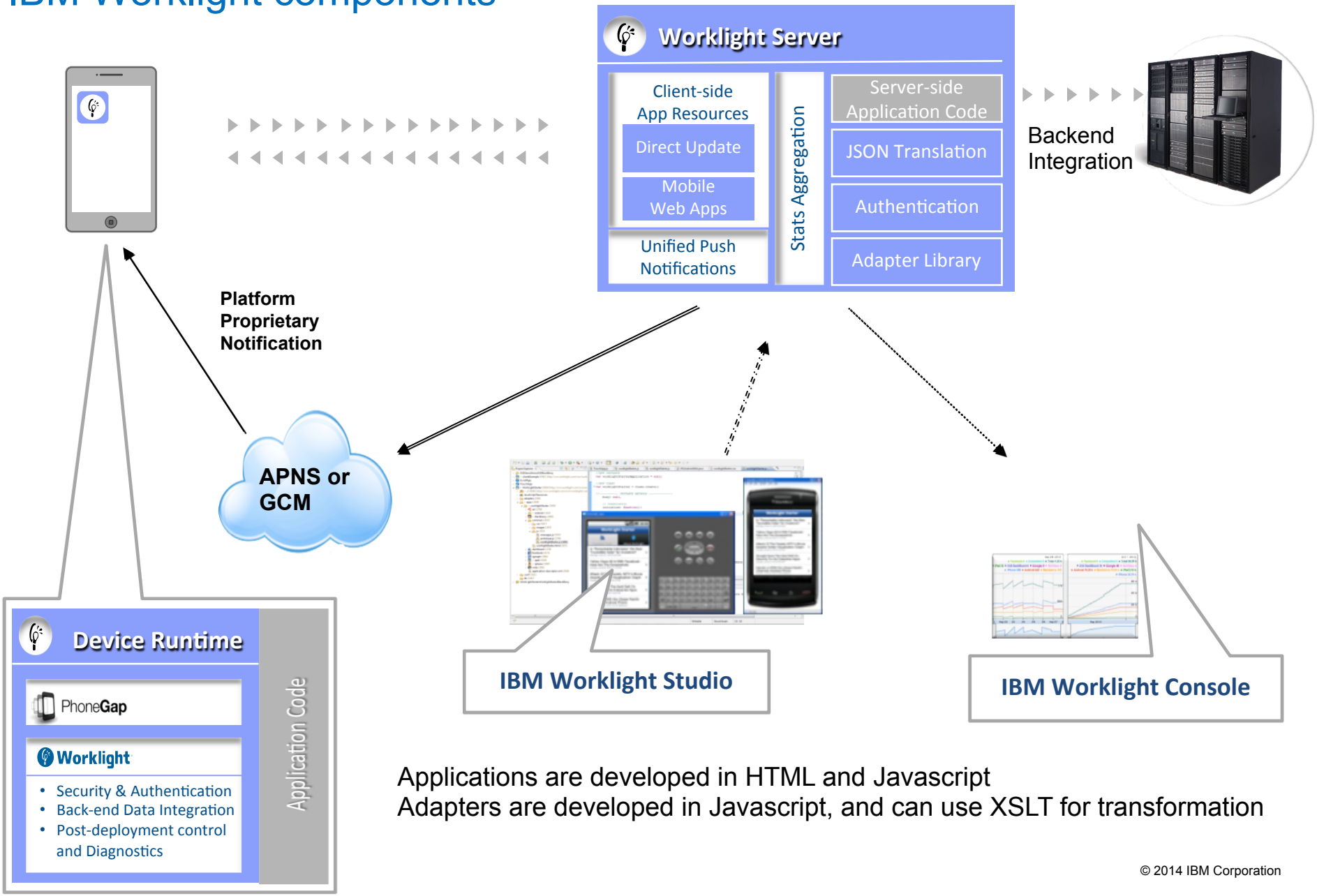
Scale & Latency

How to deliver the responsiveness that Mobile users expect at Internet scale?



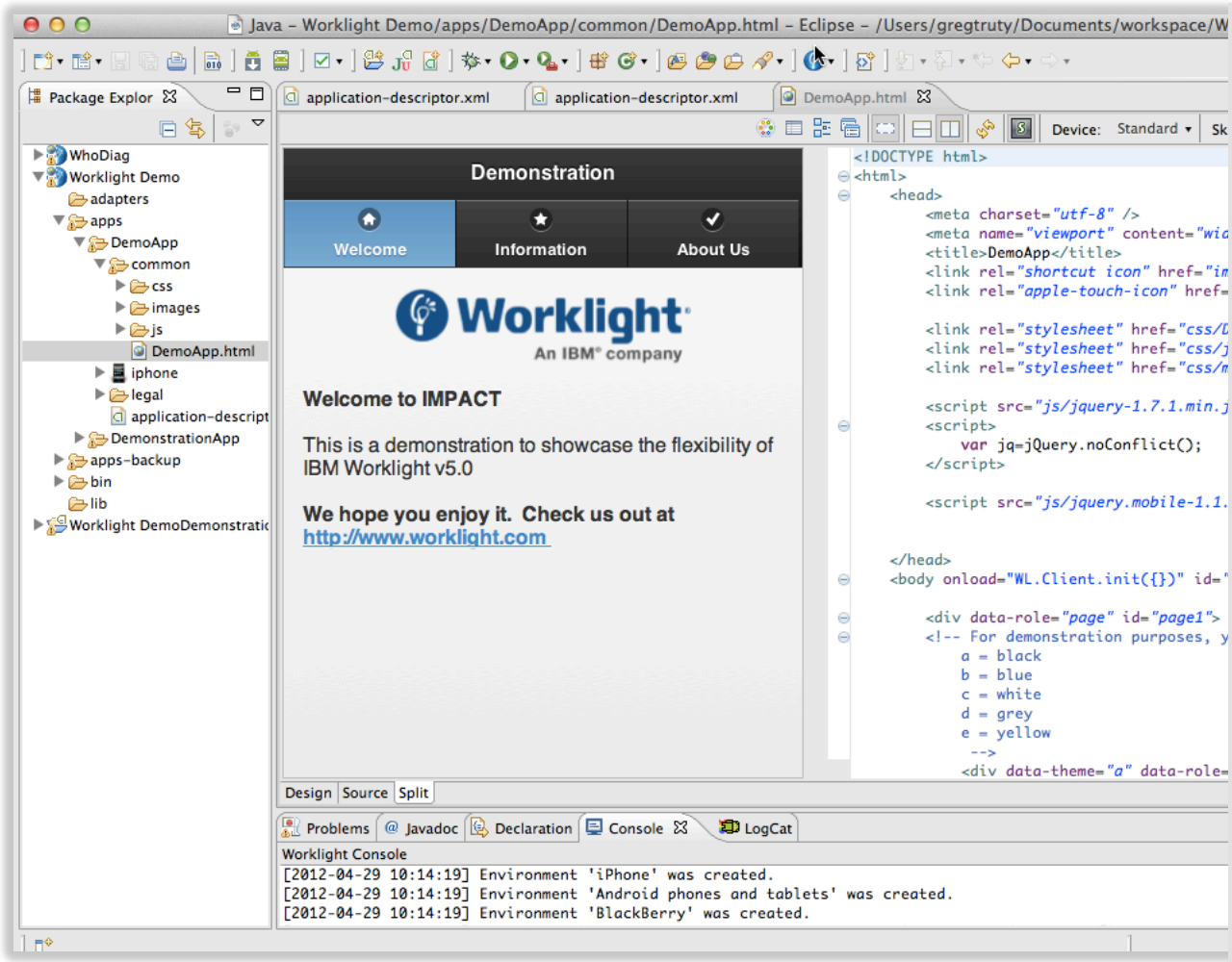
Introducing the Worklight Development lifecycle

IBM Worklight components



Applications are developed in HTML and Javascript
Adapters are developed in Javascript, and can use XSLT for transformation

IBM Worklight Studio



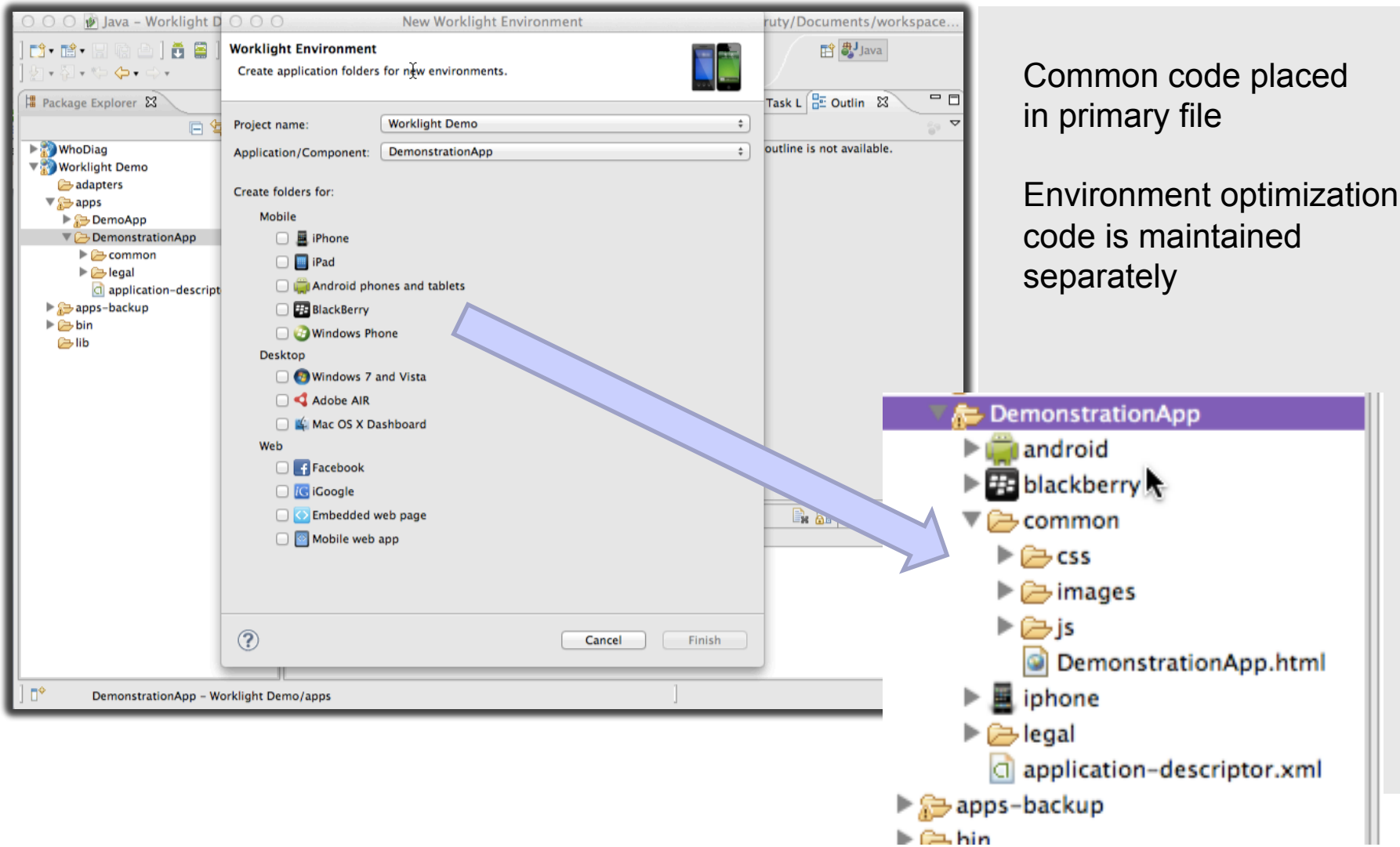
Integrated Development Environment (Eclipse Plug-in)

Application development using native and/or familiar web technologies:

- HTML5
- CSS3
- JavaScript

Integrated device SDKs allow direct access from within the IDE to emulators and code debugging utilities

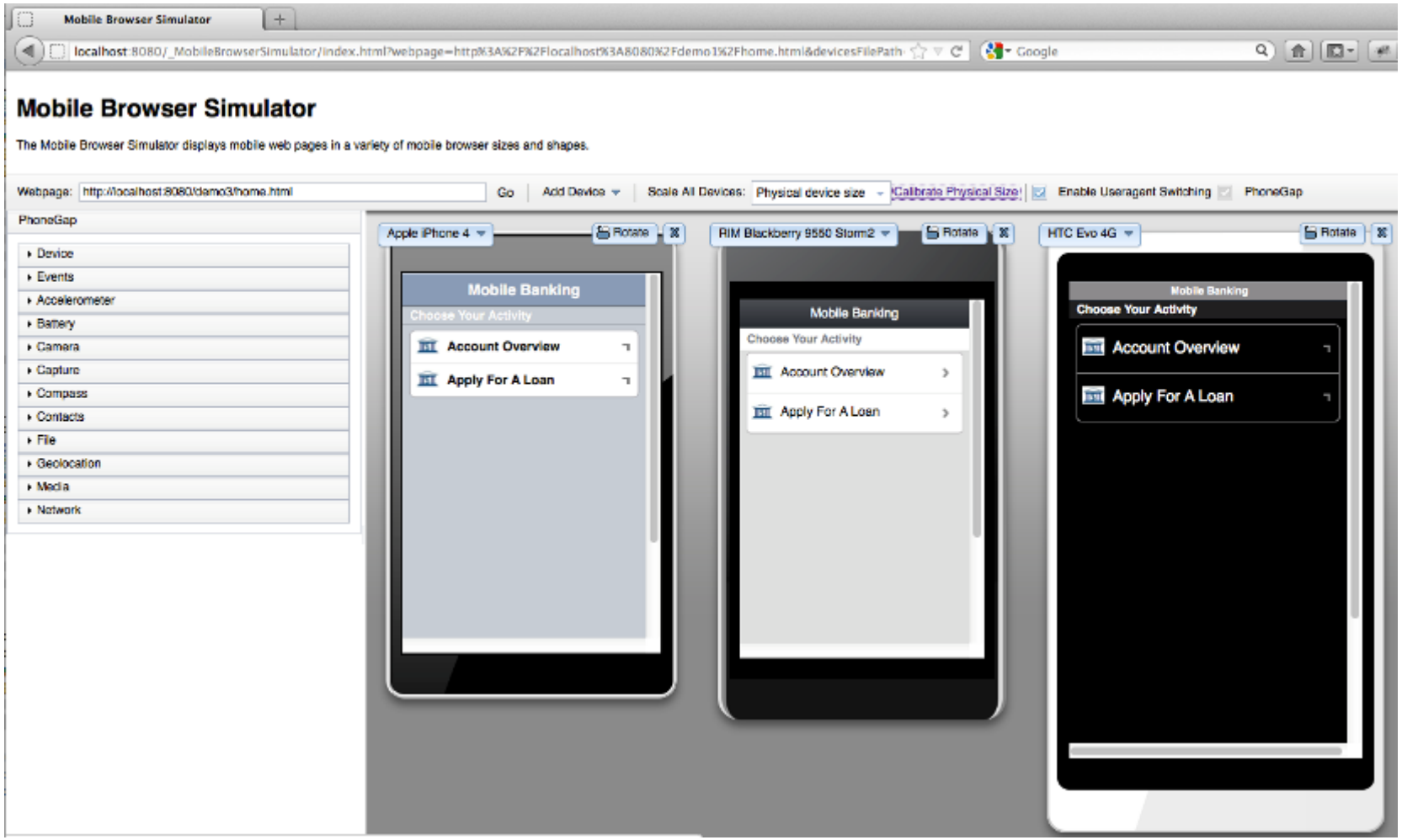
Single Shared Codebase



Common code placed in primary file

Environment optimization code is maintained separately

Preview in browser



Perform device specific tests in the Mobile Browser Simulator: supports Cordova and IBM Worklight client API

Agenda

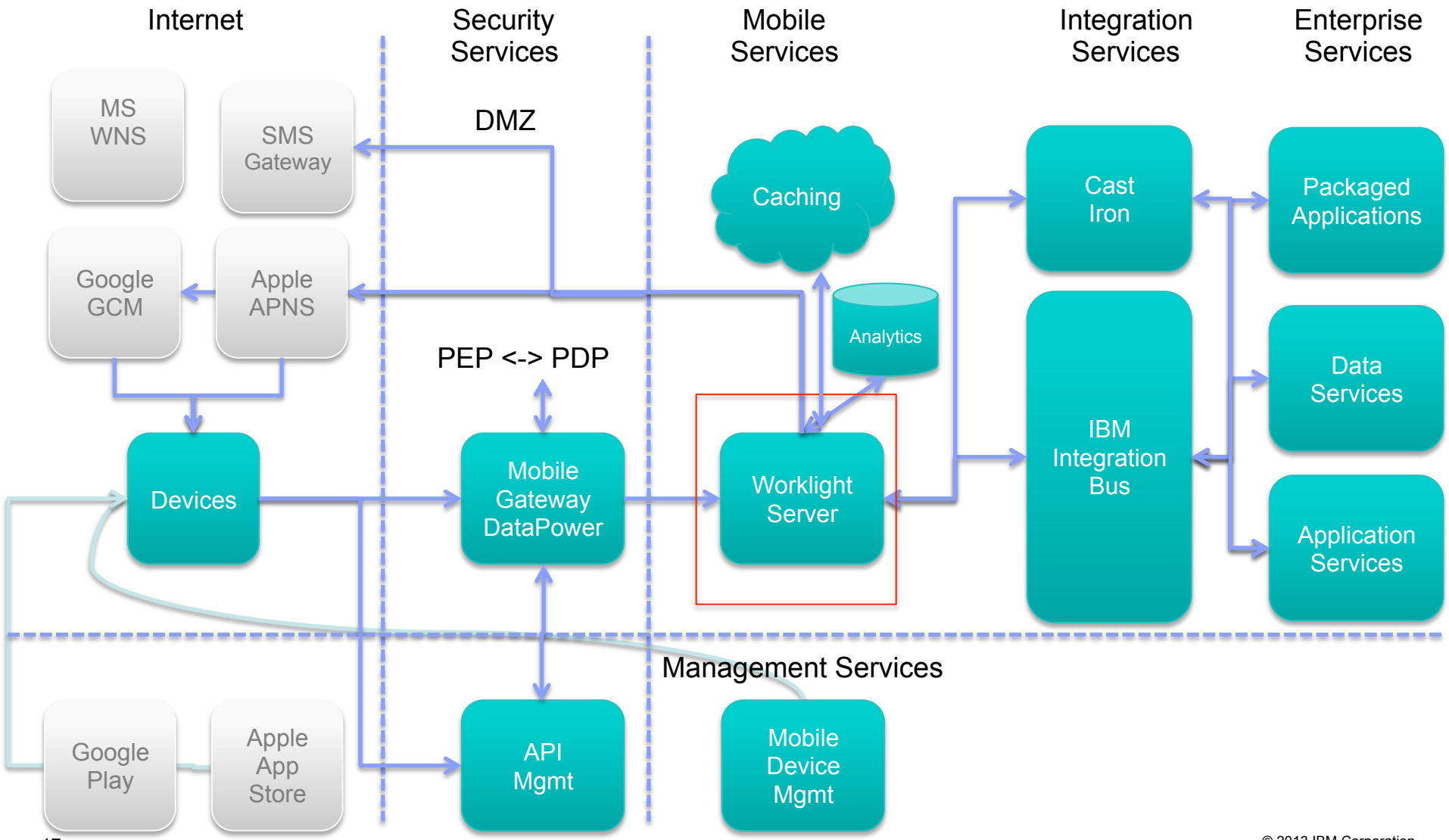
- Mobile apps and the enterprise
- **Integrating mobile apps with Enterprise Applications**
- Mobile apps and IBM Messaging
- Summary

Worklight Demo – let’s do some integration

Worklight Adapters

- Adapters provide the glue between Worklight and back-end applications
  - Provides the extensibility mechanism for Worklight to call out to back-end systems
- Adapters are invoked from mobile applications using HTTP/JSON
  - This makes Worklight adapters easy to test using web browsers
- Worklight has built-in interfaces that adapters can use (HTTP, SQL and Cast Iron)
  - Worklight has client-side JavaScript APIs so that applications can invoke services
  - Likewise, server-side JavaScript APIs are available to implement procedures (adapters)

Worklight and integration with the enterprise



How does the Worklight server fit into a mobile integration strategy?

Benefits

- Optimise the amount of data for an unstable, lower-bandwidth channel
- Optimise number of requests which otherwise increase latency over mobile network
- Manage rapidly changing app-specific services
- Productivity and agility
- Integrated security lifecycle
- App-specific server code (offload from client)

Challenges

- You're writing code!!!

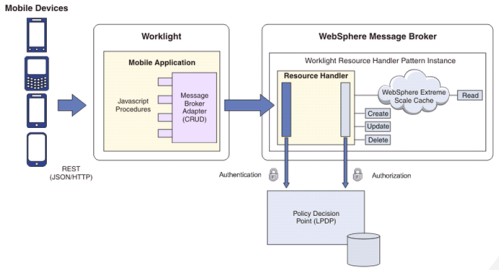
Delegate integration heavy-lifting to integration services, such as IBM Integration Bus:

- Legacy integration
- Service virtualisation
- Data and protocol transformation
- Transactionality

http://ibm.biz/WhyMobileMiddleware

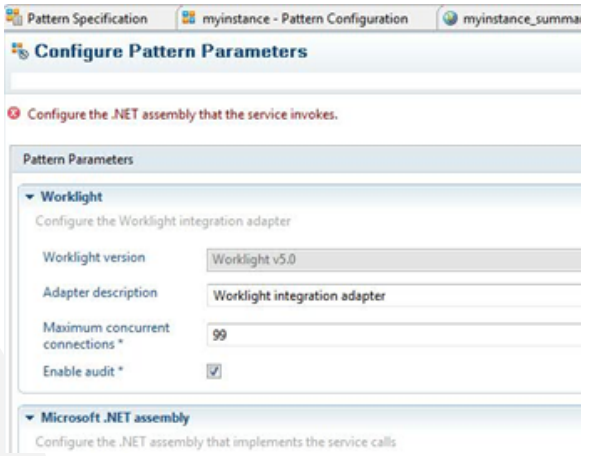
Mobile-enable your Enterprise

Accelerate access to enterprise applications, systems and data from mobile devices



Choose: Select your pattern

Configure: Accept default values or tailor for your scenario



1

2

IIB Patterns are configurable templates for common integration scenarios

Inform mobile users of key information with push notifications

Realize secure and scalable access to backend services with elastic caching

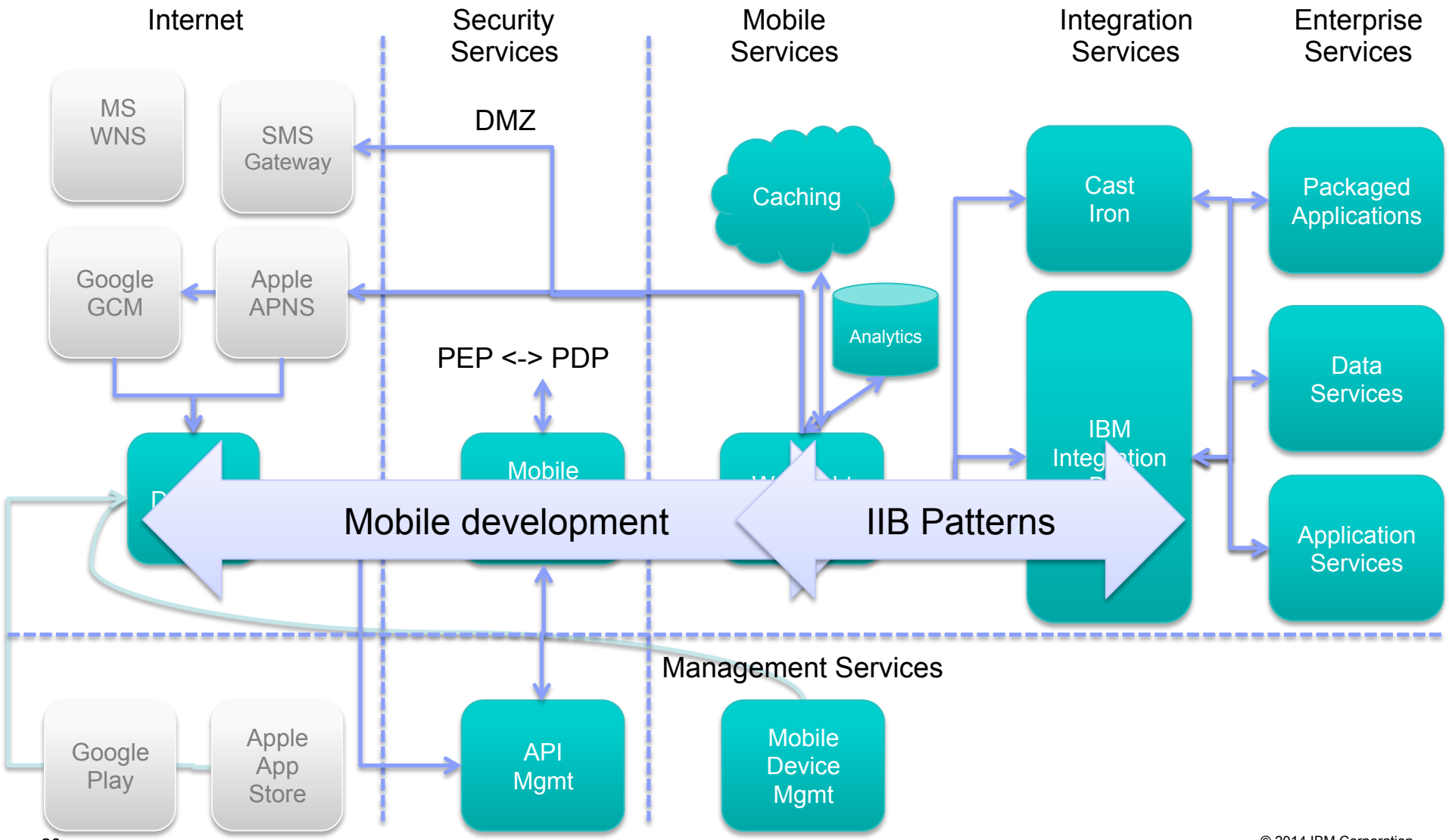
4

3

Write: Use Worklight studio – write once, run anywhere

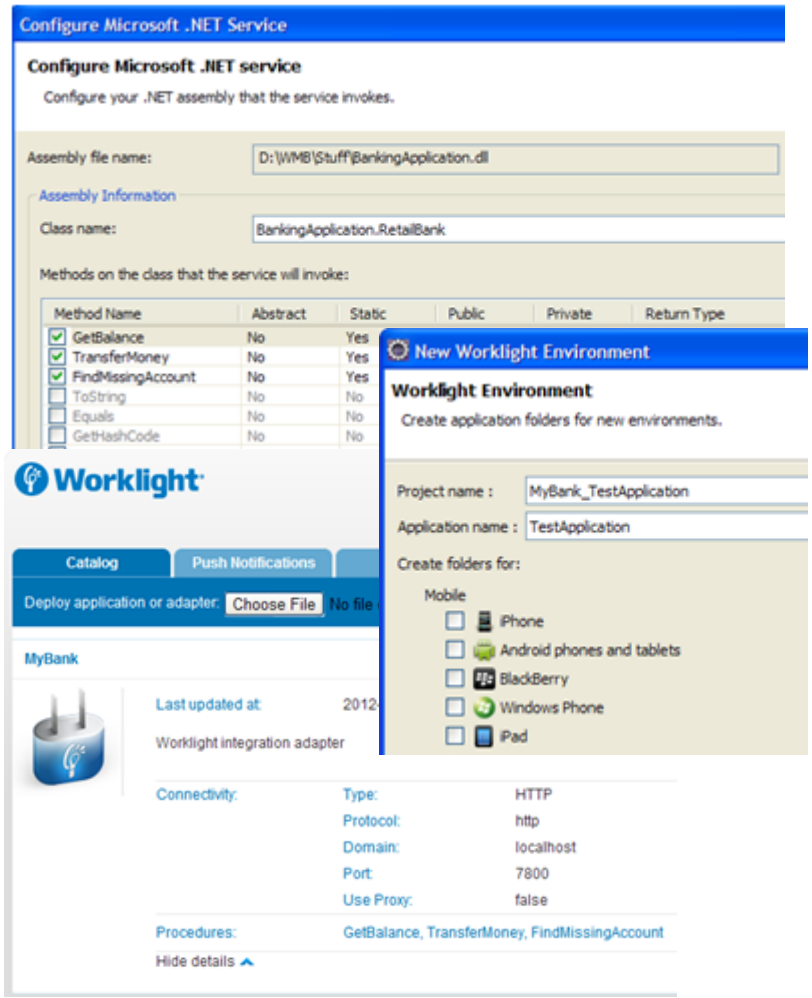
Integrate: Generate Worklight adapter ready for deployment

Worklight and integration with the enterprise



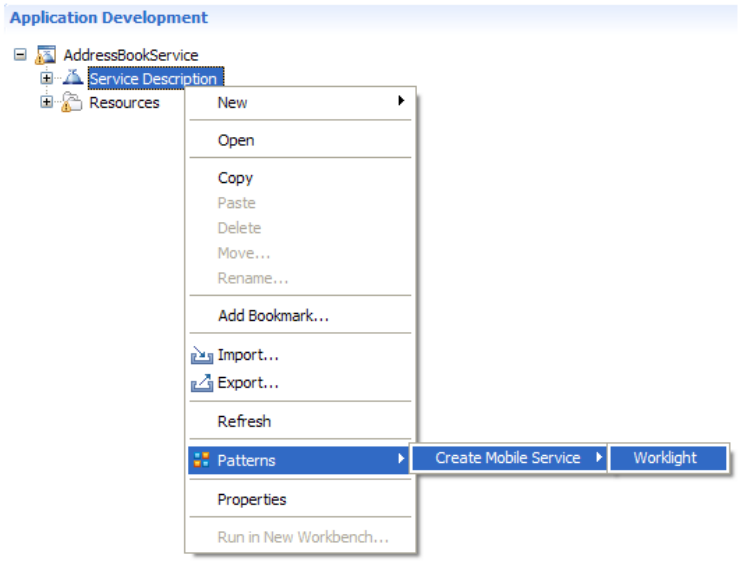
Pattern 1 - Expose a Microsoft .NET application as a mobile service

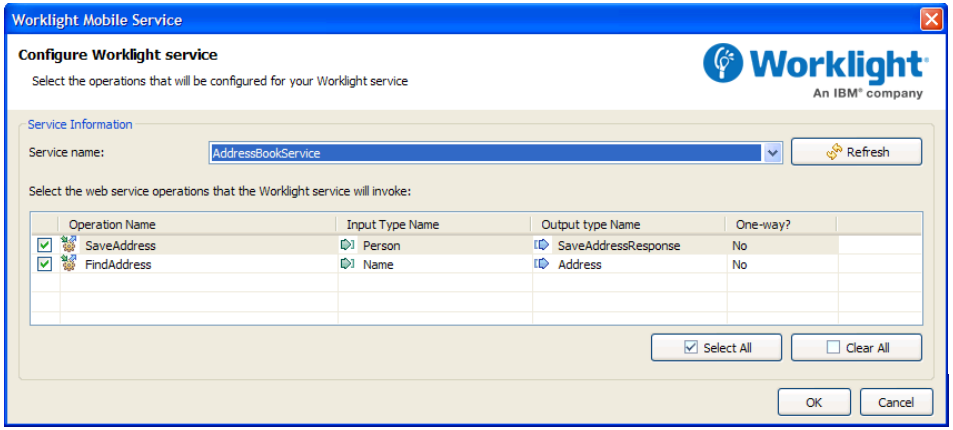
- **Simple to configure** – Drag and drop .NET assembly and enter Worklight adapter details
- **Super quick** – Pattern does all the hard work in less than a minute, generating…
  - A Web service implementation exposing desired operations
  - An adapter **ready for deployment to IBM Worklight Server**
  - A sample mobile application for **easy testing**
    - Optimised for small screen mobile devices; easy to add extra environments for iOS, Android and many more!

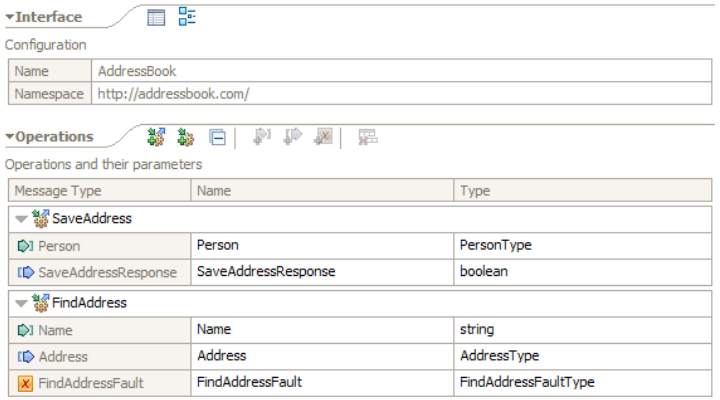


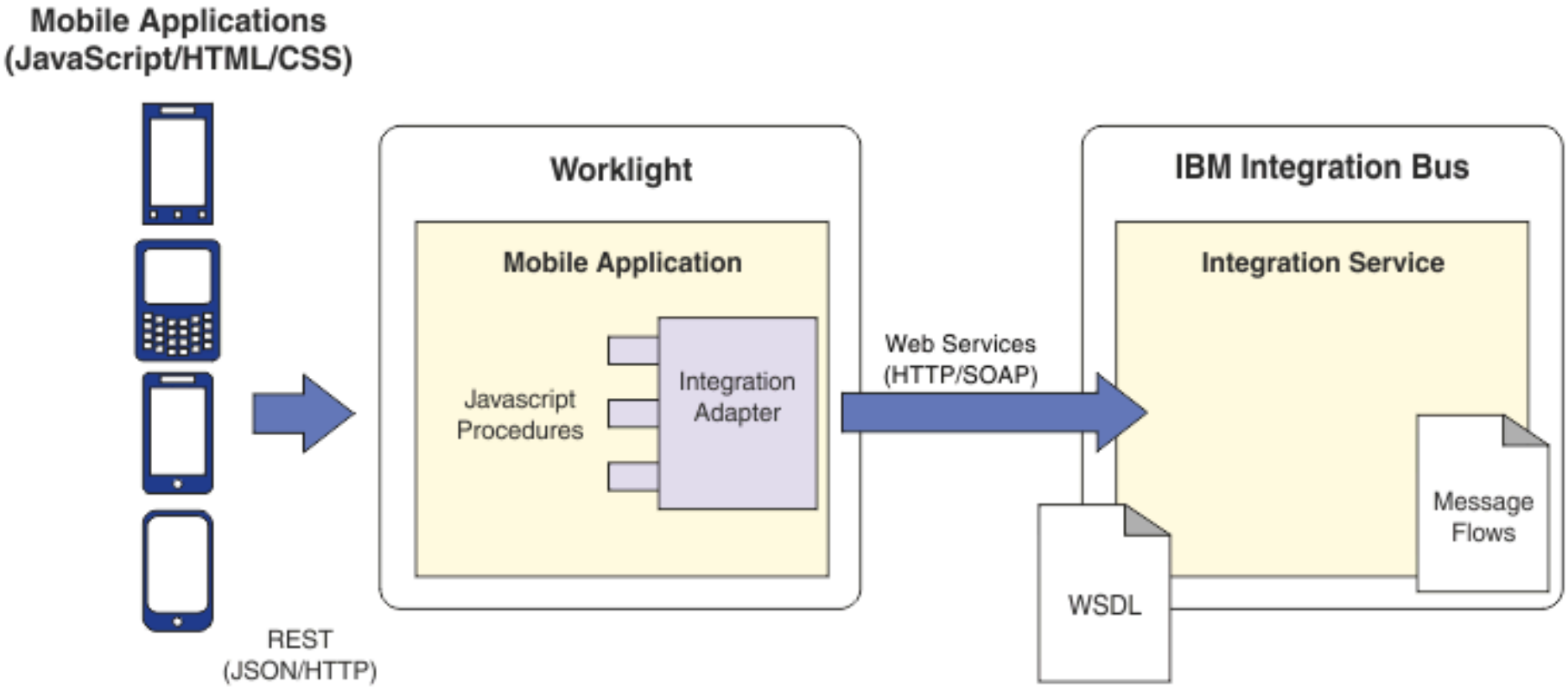
Pattern 2 - Mobile enable ANY enterprise service

- **As few as 2 Clicks!**
  - Right-click on any enterprise service (MQ, Database, Web service, CICS, IMS, etc…)
  - Left-click to create mobile service with default options
- Pattern supports extra options…
  - Choose operations to be available to mobile applications
  - Enable auditing of service requests



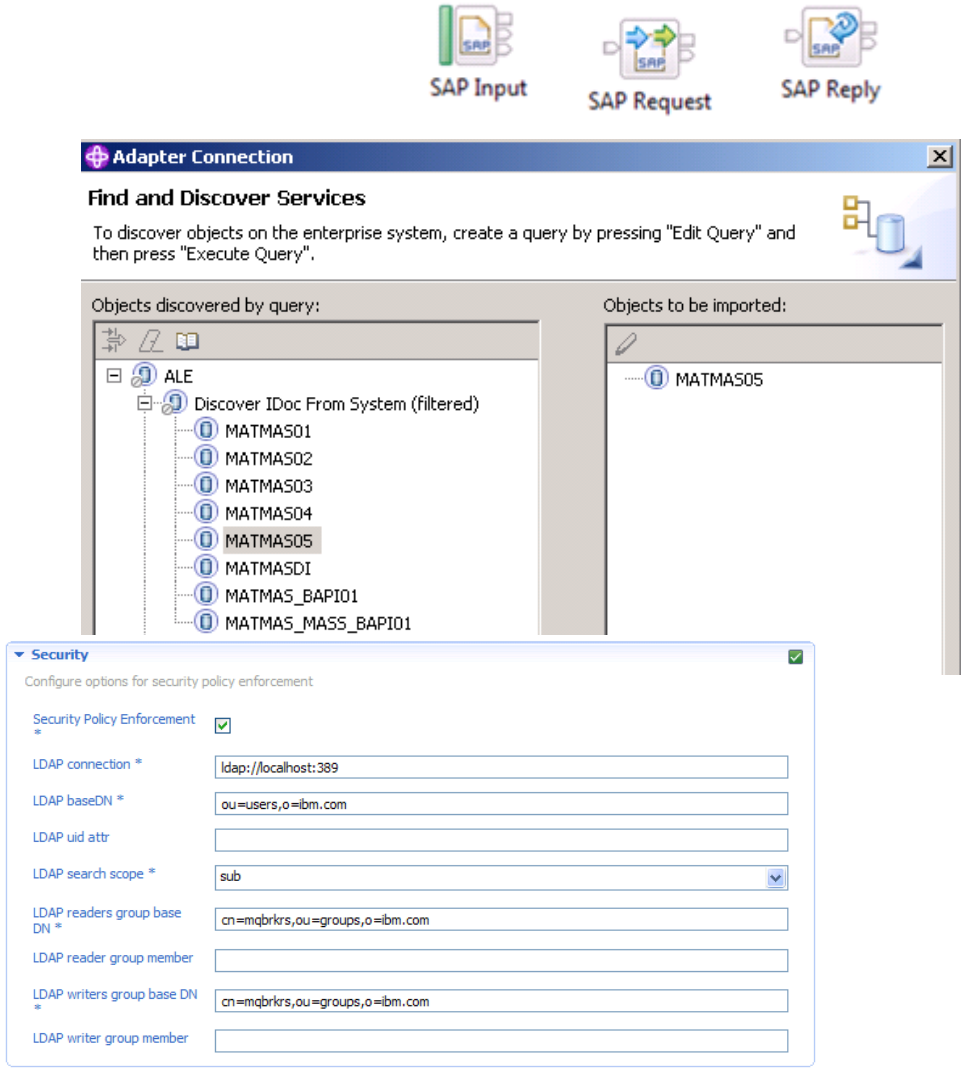


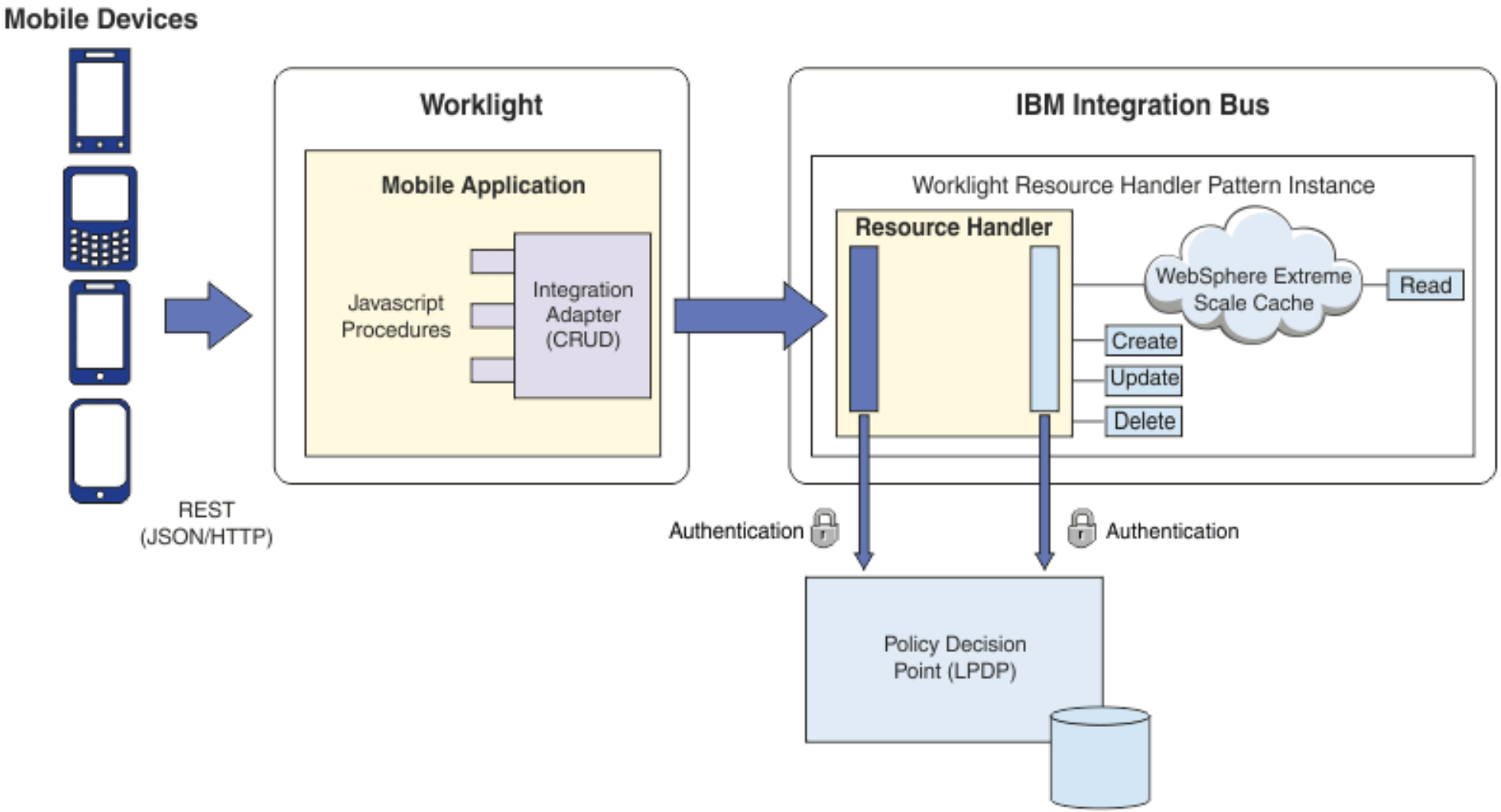




Pattern 3 - Allow mobile apps controlled access to enterprise data

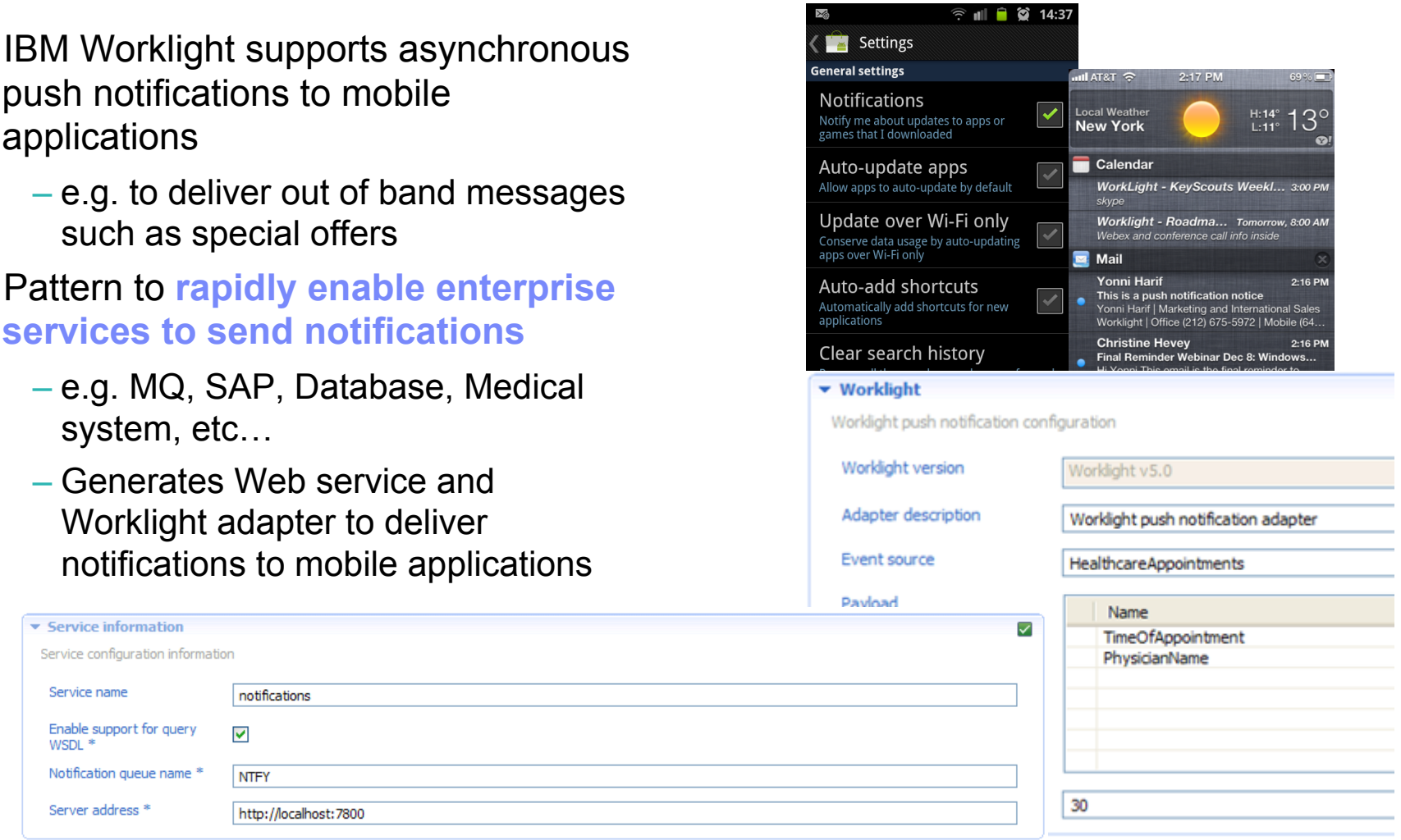
- Simple to enable mobile applications to **Create, Read, Update and Delete** enterprise data
  - Message Broker has excellent support for a wide range of enterprise applications (SAP, Siebel, JDEdwards, PeopleSoft etc…)
  - Pattern generates Worklight adapter and stubs for implementing CRUD operations
- **Quickly configure security policy** to authorize and authenticate access via external LDAP provider
- **One click to cache read resources** in IBM WebSphere Extreme Scale
  - High performing data access crucial for large volumes of mobile devices

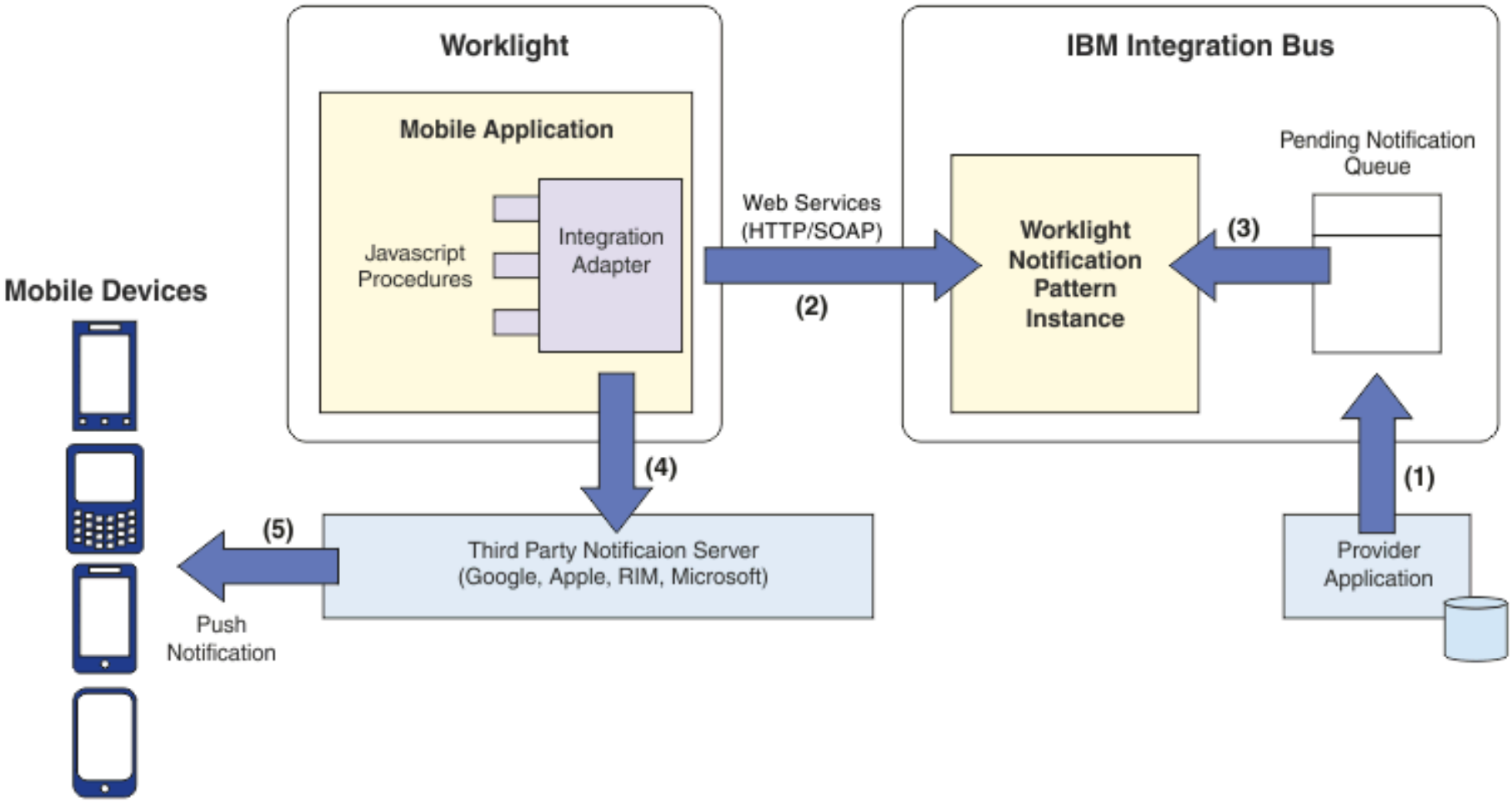




Pattern 4 - Push notification to mobile apps from within the enterprise

- IBM Worklight supports asynchronous push notifications to mobile applications
  - e.g. to deliver out of band messages such as special offers
- Pattern to **rapidly enable enterprise services to send notifications**
  - e.g. MQ, SAP, Database, Medical system, etc…
  - Generates Web service and Worklight adapter to deliver notifications to mobile applications





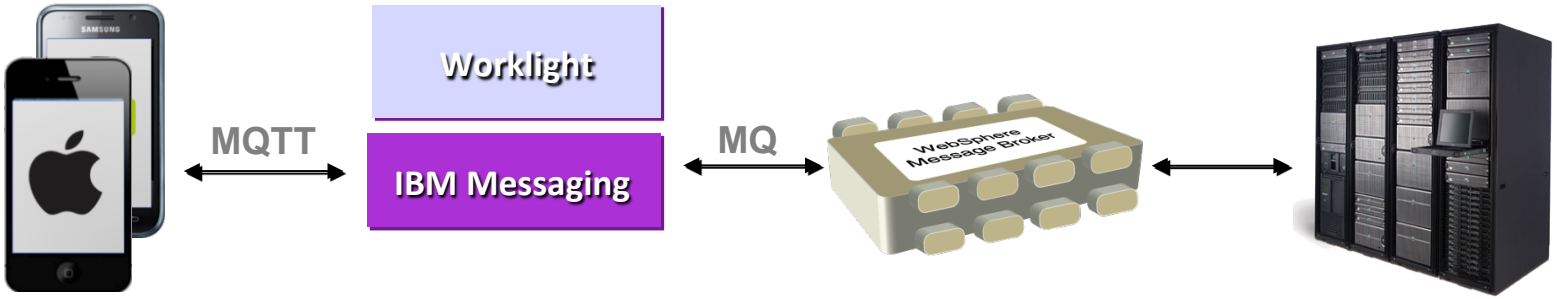
Agenda

- Mobile apps and the enterprise
- Integrating mobile apps with Enterprise Applications
- **Mobile apps and IBM Messaging**
- Summary

Key challenges for Mobile apps

- Volume (cost) of data being transmitted (especially in M2M with limited data plans)
- Power consumption (battery powered devices)
- Responsiveness (near-real time delivery of information)
- Reliable delivery over fragile connections
- Security and privacy
- Scalability

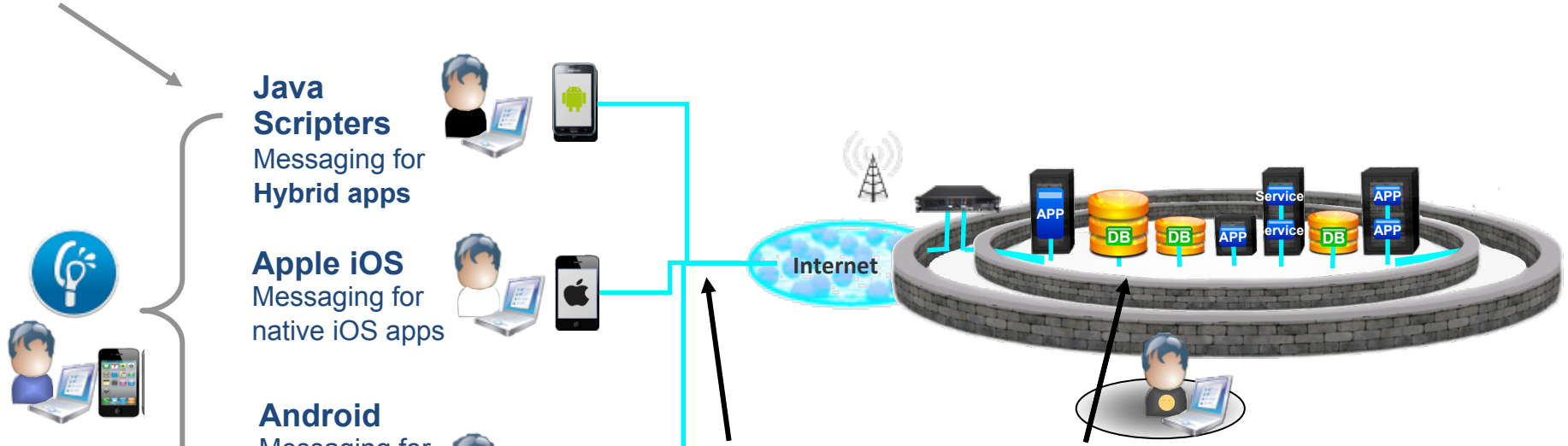
Mobile message exchange patterns – beyond simple request/response



Reliable asynchronous transactions	User submits a transaction. One or more responses may come back over time.	MQTT provides **reliability** and **store/forward** of requests and responses if needed – reducing the amount of application code
Continuous update of realtime information	Server-side data is “streamed” to the device and used to update the UI. In most cases this is only required when the app is in the foreground	Small MQTT header size reduces **battery consumption** and **network traffic. One->many** publish/subscribe reduces load on application
Notification	Sending alert or other informational message to the device. The app may or may not be running at the time.	Avoidance of polling reduces **battery consumption** and **network traffic**. **Store/ forward** of important notifications if app/device is not contactable
Collection of data from device	Data sent to the server coming either from User Interface, of from onboard sensors or from devices attached to the phone	Small MQTT header size reduces **battery consumption** and **network traffic. Store/ forward** of messages. **One->many** publish/subscribe

New Messaging clients for Mobile apps

Clients for Mobile and M2M Messaging
•Including Simple Javascript Messaging API

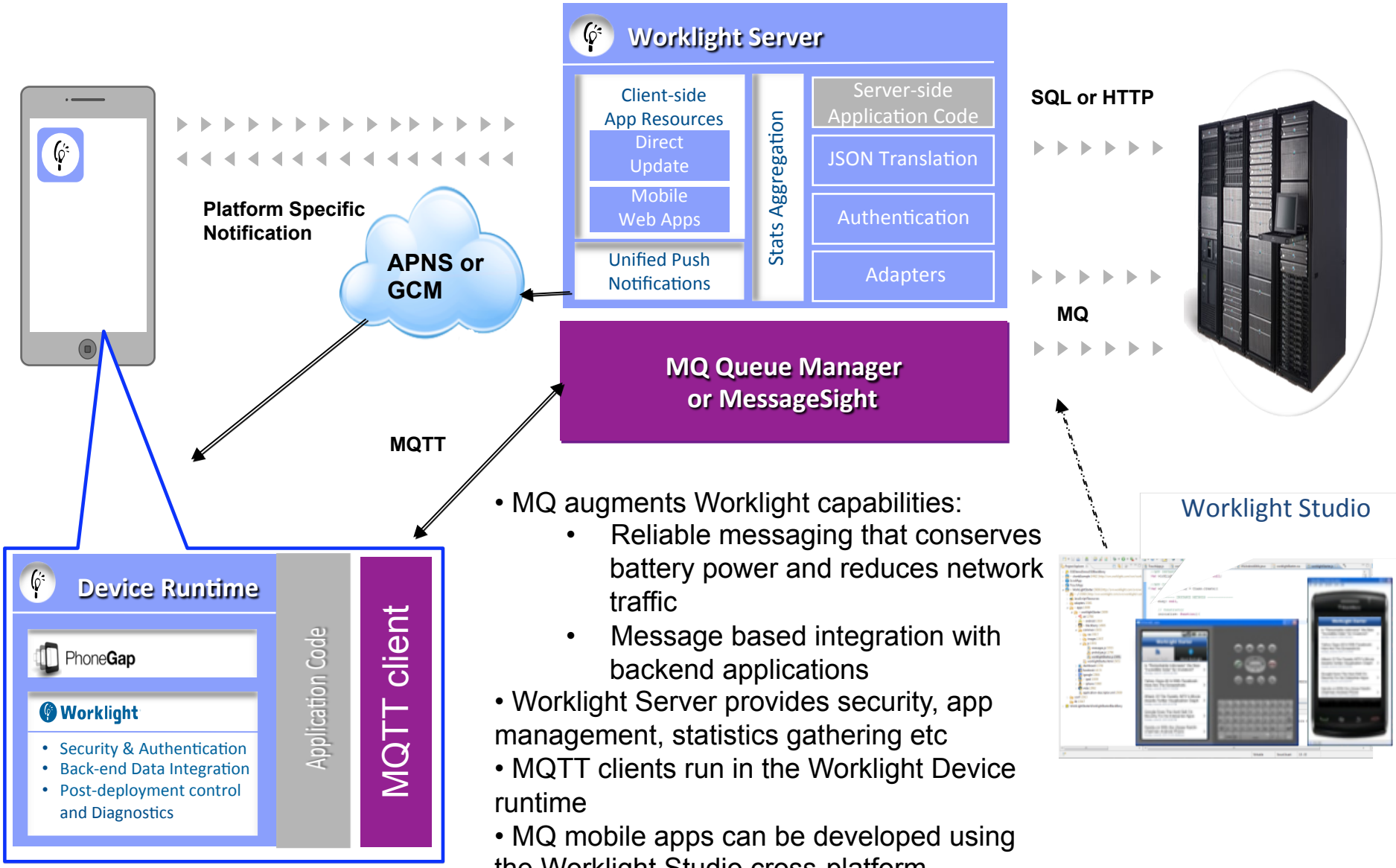


Worklight developers
Developer experience augmented with mobile messaging

MQTT
- Reliable messaging protocol
- Access to enterprise
- Reliable delivery
- Conserve device power
- Reduce network traffic

ESB and MQ messaging
- Universal Enterprise Messaging
- Provide access to enterprise apps and data already connected by MQ
- Pub/sub model to enable dynamic distribution of notifications

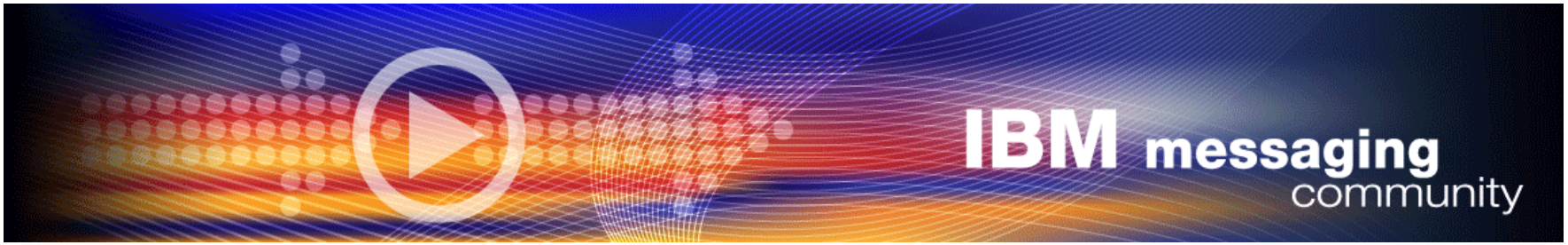
MQ and Worklight



- MQ augments Worklight capabilities:
  - Reliable messaging that conserves battery power and reduces network traffic
  - Message based integration with backend applications
- Worklight Server provides security, app management, statistics gathering etc
- MQTT clients run in the Worklight Device runtime
- MQ mobile apps can be developed using the Worklight Studio cross-platform development tool

New developerWorks messaging community

- **Objective**
  - A single place for developers wishing to create mobile messaging applications
- **What is being delivered?**
  - MQ Mobile Client Pack, providing enhanced MQTT messaging clients for mobile devices and sensors
    - Ease of integration with enterprise applications
    - One-to-many message delivery (publish/subscribe)
    - Reliable delivery over fragile connections
    - Near realtime push of data from the server
    - Low power consumption & scalability
  - Articles, code samples and sample mobile applications
- **How available?**
  - Downloadable from this new Messaging community
  - Clients are fully supported when used with relevant IBM products, for no extra charge



Agenda

- Mobile apps and the enterprise
- Integrating mobile apps with Enterprise Applications
- Mobile apps and IBM Messaging
- Summary