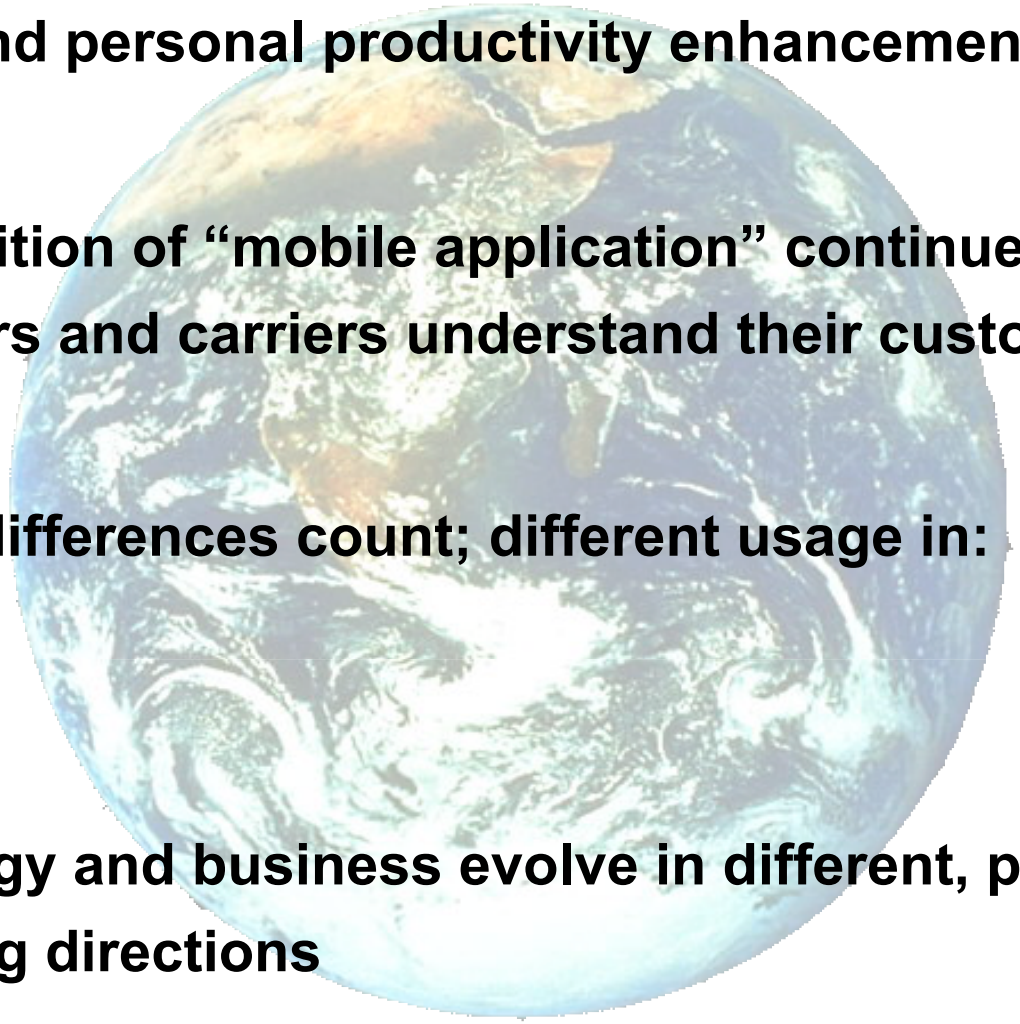
The Mobile Java Continuum

Eugene Ciurana
eugenex@walmart.com
Chief Architect, Advanced Technology Group
Walmart.com

Mobile Java Applications

- **The elusive killer app**
  - **Lots of mobile applications appear every year**
  - **There is no mobile Java killer app...**
- **Why?**
- **The main reason: mobile Java applications should be thought as part of a continuum, not as an end**
- **J2ME is only part of the continuum**
  - **The application must integrate with a supporting back-end**
  - **May have to interact with heterogeneous technologies from multiple vendors**
  - **A carrier or device manufacturer may need to be involved**

Mobile Java Applications



- Games and personal productivity enhancements are the norm
- The definition of “mobile application” continues to evolve as vendors and carriers understand their customer’s needs better
- Cultural differences count; different usage in:
  - Japan
  - Europe
  - US
- Technology and business evolve in different, perhaps conflicting directions

Mobile Java Applications

- **So this presentation is about design, creation, and deployment issues...**
  - **Environments**
  - **Technologies involved**
  - **Caveats**
- **This includes:**
  - **MIDP 2.0/CLDC 1.1**
  - **Availability of JSR-compliant vendor APIs**
  - **Internet access for mobile devices**
  - **Mobile carriers and other Internet providers**
- **For this discussion, we assume that carriers are US companies**

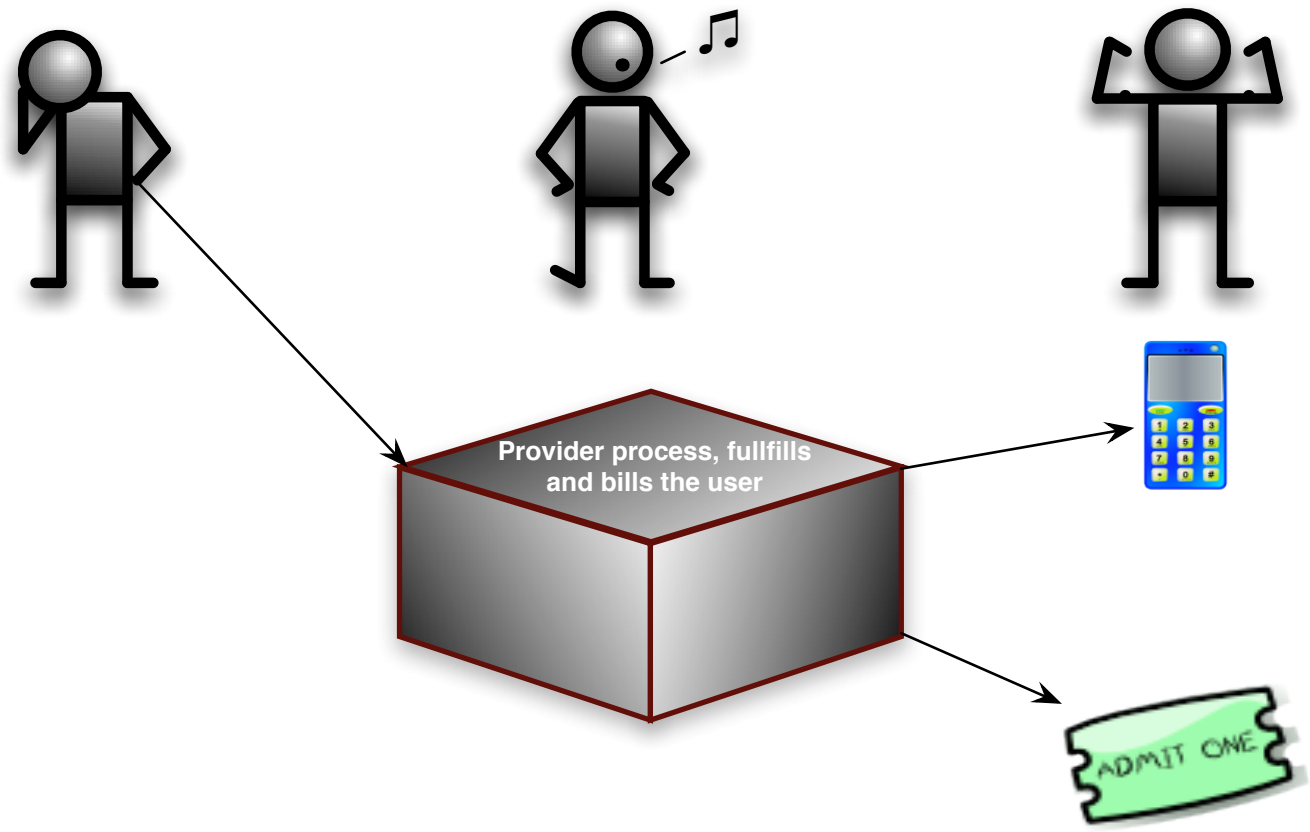
Architecture: Mobile Applications

Mobile applications: products or services delivered through user interactions with a mobile device

- **Closed loop - requests are prepared from, and fulfilled to, the mobile device itself**
  - **Ringtones**
  - **Media subscription services**
- **Open loop - requests are prepared from the mobile device and fulfilled on a different device, medium, or context**
  - **Tickets**
  - **Photo prints**

Architecture: Mobile Applications

What the User Experience is Like...



Architecture: Mobile Applications

What is really involved in the user experience:

- **A mobile device with a common transport protocol**
  - **Normally TCP/IP encapsulated through GPRS, BREW, etc.**
  - **Some times this could be simple SMS**
- **A gateway that enables communications with the product or service provider**
  - **Carriers love to play this part and collect a toll**
- **One or more service providers**
  - **One per application or service**
- **One or more fulfillment servers associated with each provider**

End-user Experience Considerations

- **PDAs and phones are naturally interactive devices**
- **Issues:**
  - **Small screens**
  - **Awkward input devices**
  - **Limited resources (primary and secondary memory, others)**
  - **At the same time, they're overloaded with features!**
    - **Cameras**
    - **Media players**
- **The OS offers other opportunities and limitations**
  - **PalmOS (will it be defunct soon?)**
  - **Symbian**
  - **Linux**
  - **PocketPC**
  - **Others**

End-user Experience Considerations

- **The operating system and device feature set might be proprietary or require special programming**
  - **Example: background execution or downloads**
    - **The OS may support it, even when the device is closed**
    - **Device manufacturers may disable it to preserve battery life or to prevent users from running huge connection bills**
  - **Competitive advantage in not exposing the dedicated features (megapixel cameras, media players) to third parties**
- **The Java API is sandboxed within this environment**
  - **Even more limited access to device resources than what the device manufacturer provides**
  - **Technological and political restrictions also play a part**
    - **JSRs may be too vague and don't implement full features *by design***

End-user Experience Considerations

Example: Bluetooth discovery of other devices in the PAN	Example: Manipulating photos in a megapixel camera phone
Requires: JSR-82, Bluetooth / OBEX API package	Requires: JSR-75, file connection API package
Desired functions: device discovery, service discovery, offer service registration	Desired functions: root file system discovery, list files per file system, manipulate JPEG images
Reality: most J2ME/JSR-82 compliant devices do not support these features unless the JVM is running and the device is in an active state; cannot fork the JVM as a background process	Reality: file system support is not uniform across J2ME/JSR-75 devices - what works in some doesn't in others; the MIDP specification defines only PNG support as required - JPEG or other format manipulations require native device API

Target devices must be throughly tested and generally demand the carrier's or device manufacturer's assistance or approval.

Application Design

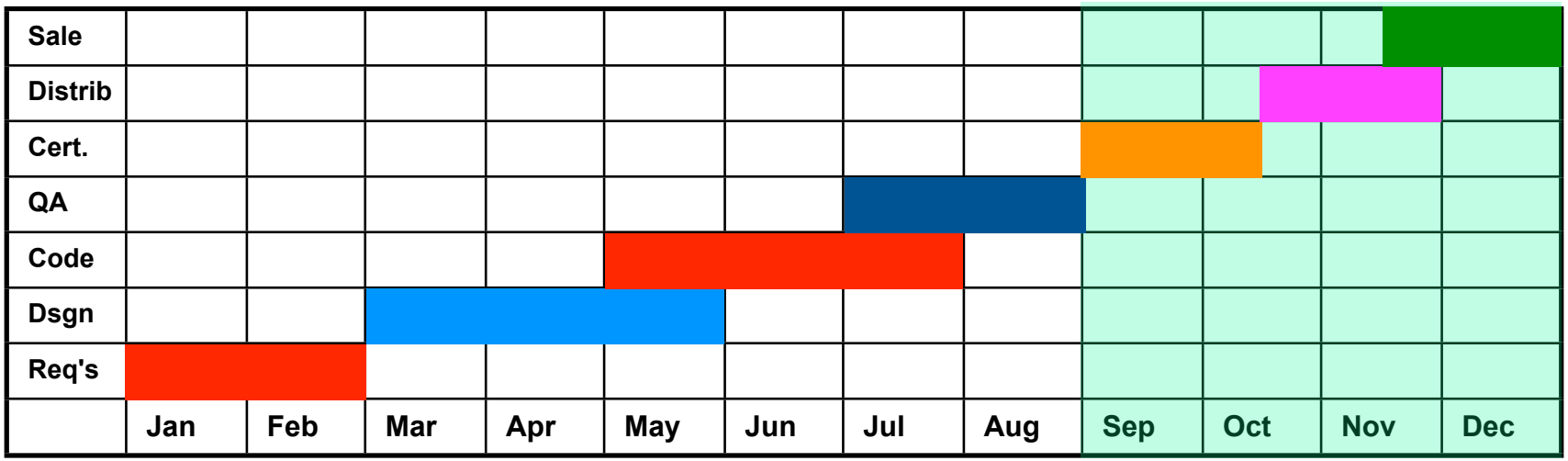
- **Enterprise applications resemble closed-loop commercial services**
  - **Data collection/validation applications**
  - **Standardized development toolset**
  - **Standardized deployment platform**
  - **They are easier to design, develop, test, and deploy because the options are limited**
- **Commercial applications**
  - **Open- or closed-loop**
  - **Usually must find a partner (device manufacturer, carrier, or both)**
  - **Assume heterogeneous devices and target the lowest common denominator**

Application Development Life Cycle

- **Similar to any other app. dev. life cycle unless...**
- **...you involve a partner.**
- **If the partner is a device manufacturer, life is a lot simpler because NDAs lubricate the process and give you access to APIs, features, etc.**
- **If the partner is a carrier, keep in mind that they will want to collect a toll for every operation performed through the device**
  - **Who bills the end-user?**
  - **What percentage of the bill goes to you?**

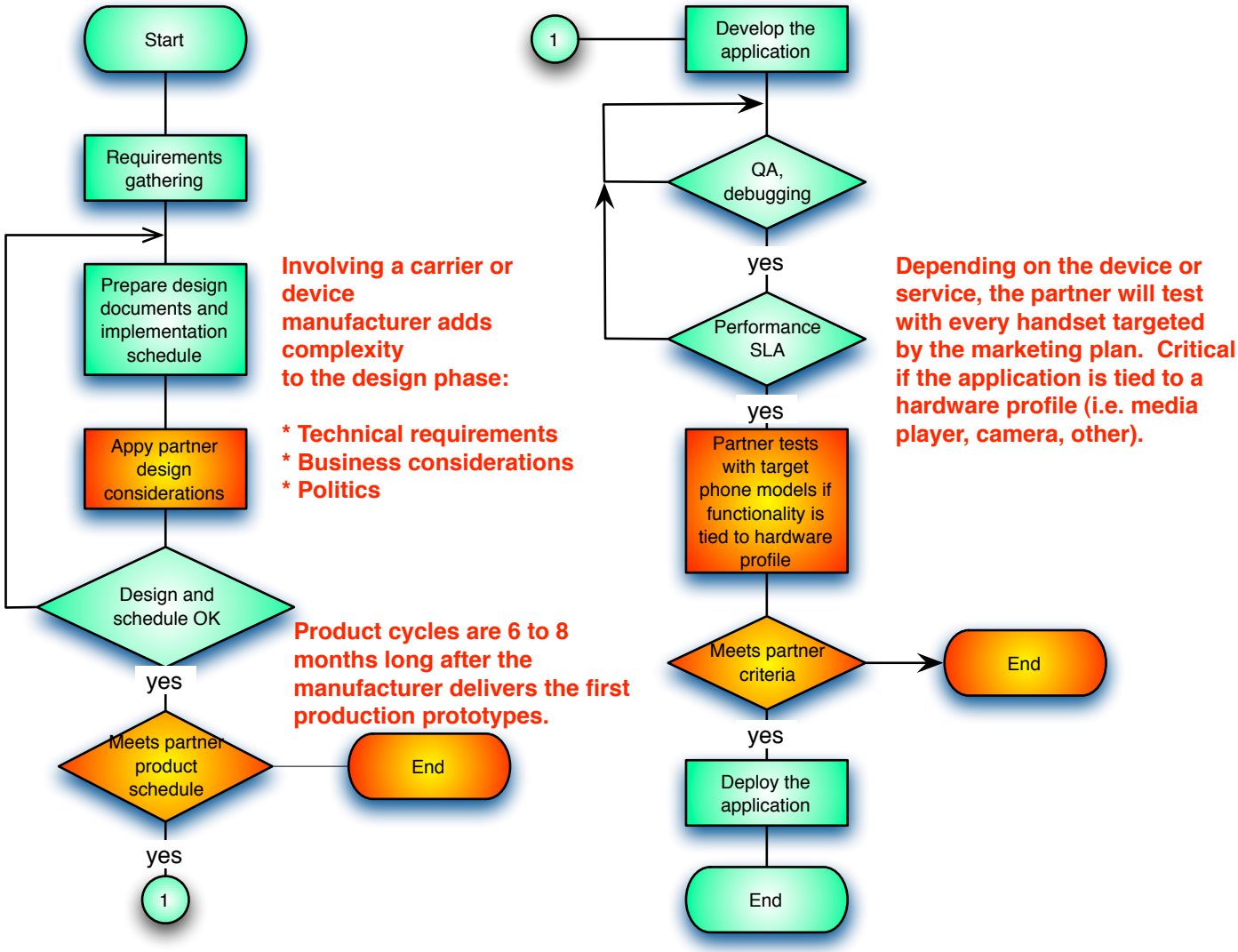
Application Development Life Cycle

Commercial application, sold through retail channels



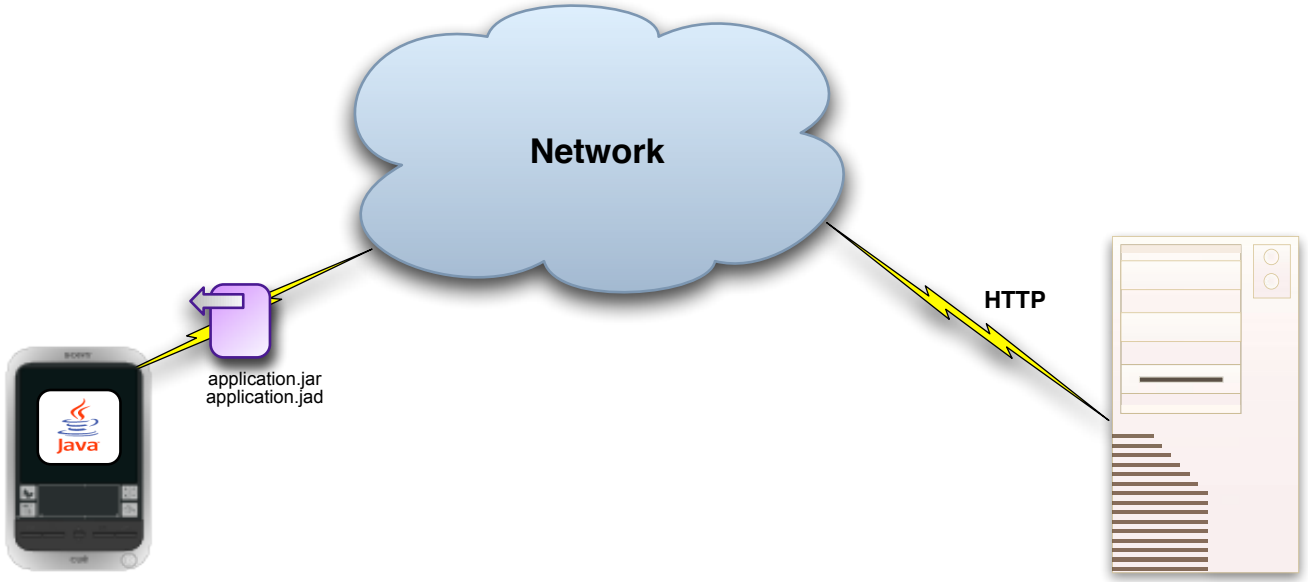
Holiday season

Application Development Life Cycle



Application Delivery

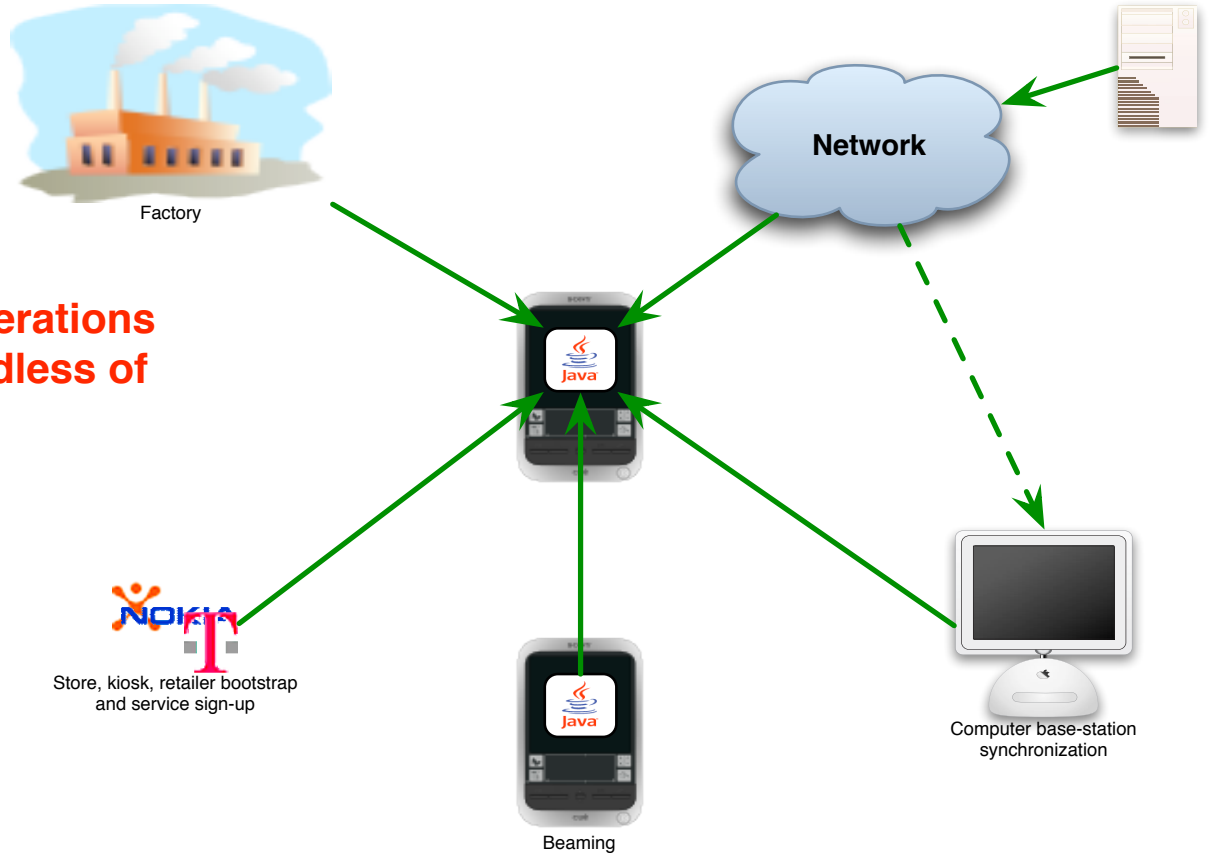
- **Delivery is simple for enterprise applications**



- **Uses standard protocols and mechanisms**
  - **application.jar - Java classes and resources**
  - **application.jad - JAr Descriptor file - metadata (vendor, size, etc.)**

Application Delivery

- **Commercial applications may have multiple delivery channels**



Business considerations may apply, regardless of technology base.

Application Support

- **Applications must be certified on all intended target platforms**
  - **Unless you want to have a headache**
- **Support the application as one-time installation**
  - **Handset and carriers like this idea - forces upgrades to gain access to new functionality**
  - **Legacy - how long will you support it?**
- **Incremental upgrades, bug fixes**
  - **Helps maintain the installed based up-to-date**
  - **QA and certification burden: regression testing becomes a problem**
    - **Whose?**
      - **Device manufacturer's, carrier's, app developer?**

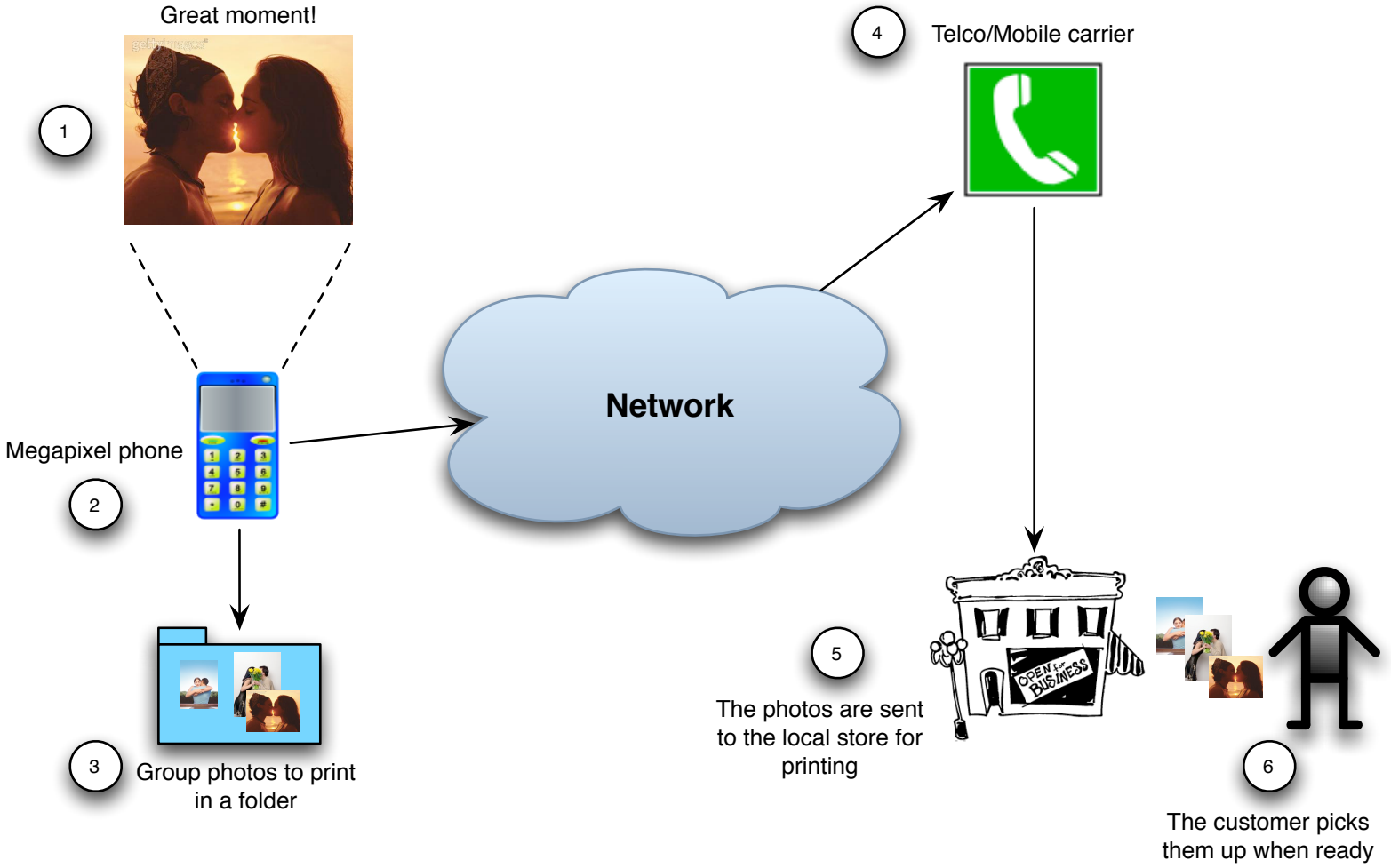
Snacking Behavior

- **User-friendliness in UI - make it easy for the user to accomplish the task in as few steps as possible**
- **Mobile devices are "impulse devices"**
  - **I want it now, I get it now**
  - **Grab the end-user quickly if your application is a service**
- **End-users show "snacking behavior"**
  - **3G video/media playback ~= 3 minutes or less**
  - **Photos: I'm here and can take a photo!**
  - **Ringtones: fetch and install in a single SMS interaction**
  - **Music: for practical purposes, the iPod ROKR and similar devices are a bust**
    - **They allow music deployment only through a docking station**

A Real-Life Application: Photo Prints

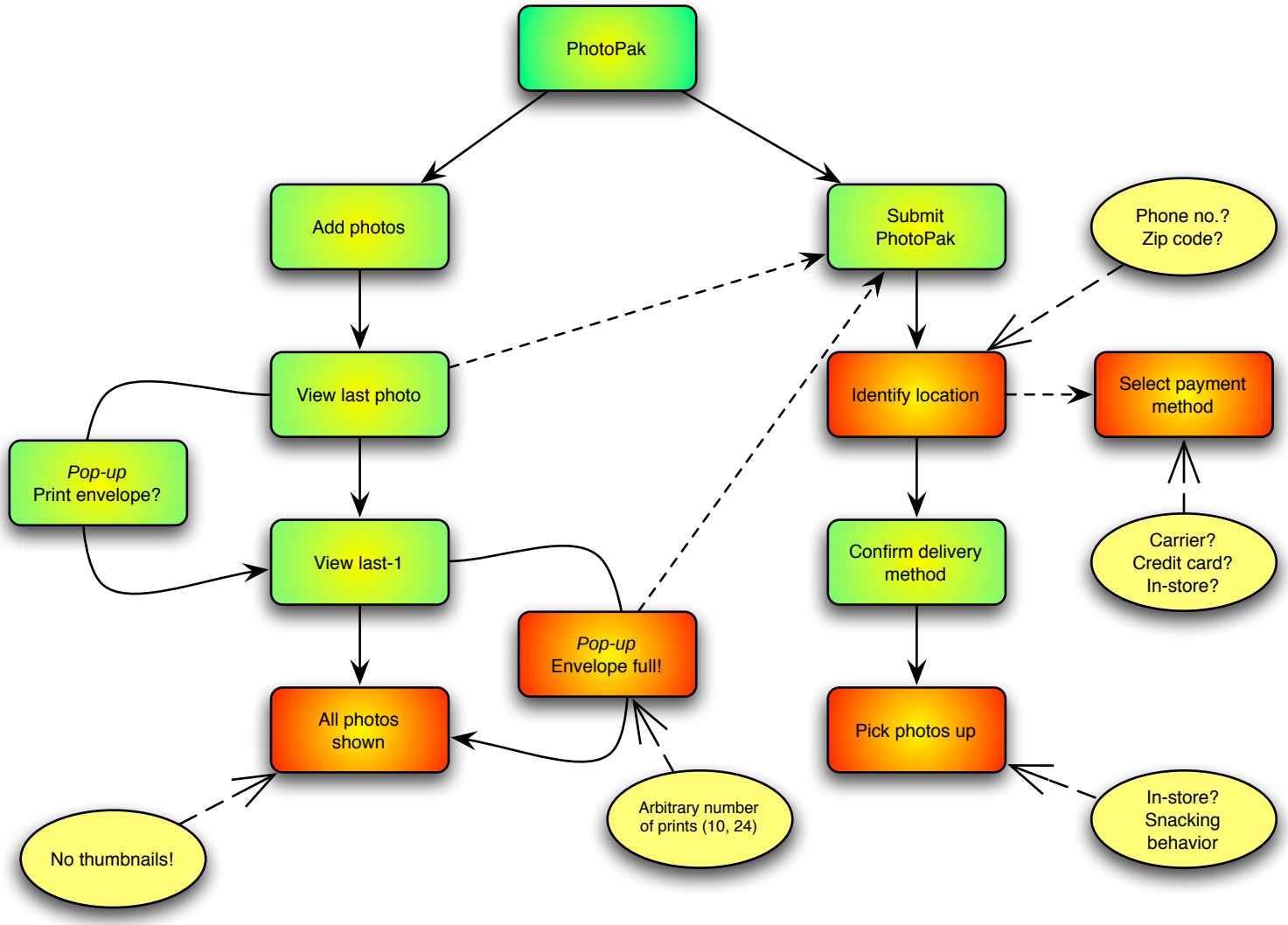
- **Camera-enabled mobile devices provide convenience**
  - **Many have megapixel capabilities**
- **How do you print the photos?**
  - **Download the images to a computer, then order**
  - **Write the photos to an SD card and print from an in-store kiosk**
- **Imagine a world where you can take your photos and order prints from your phone**
  - **Convenience**
  - **Snacking behavior**
- **How would you implement this?**

A Real-Life Application: Photo Prints



A Real-Life Application: Photo Prints

Photo Processing Java Application Flow



Real-Life Application Considerations

- **Define a default fulfillment location?**
  - **Based on user profile**
  - **Based on phone number**
  - **Based on geographical location**
- **Is it necessary to tie this workflow with other applications?**
  - **When transferring the package for processing, update a user's web site account (i.e. web photo site, retailer web site, tickets, what?)**
- **How are end-user accounts created or linked?**
- **Will the carrier allow third parties devices to access the application and service?**
  - **Politics**
  - **Support**

Real-Life Java Application Considerations

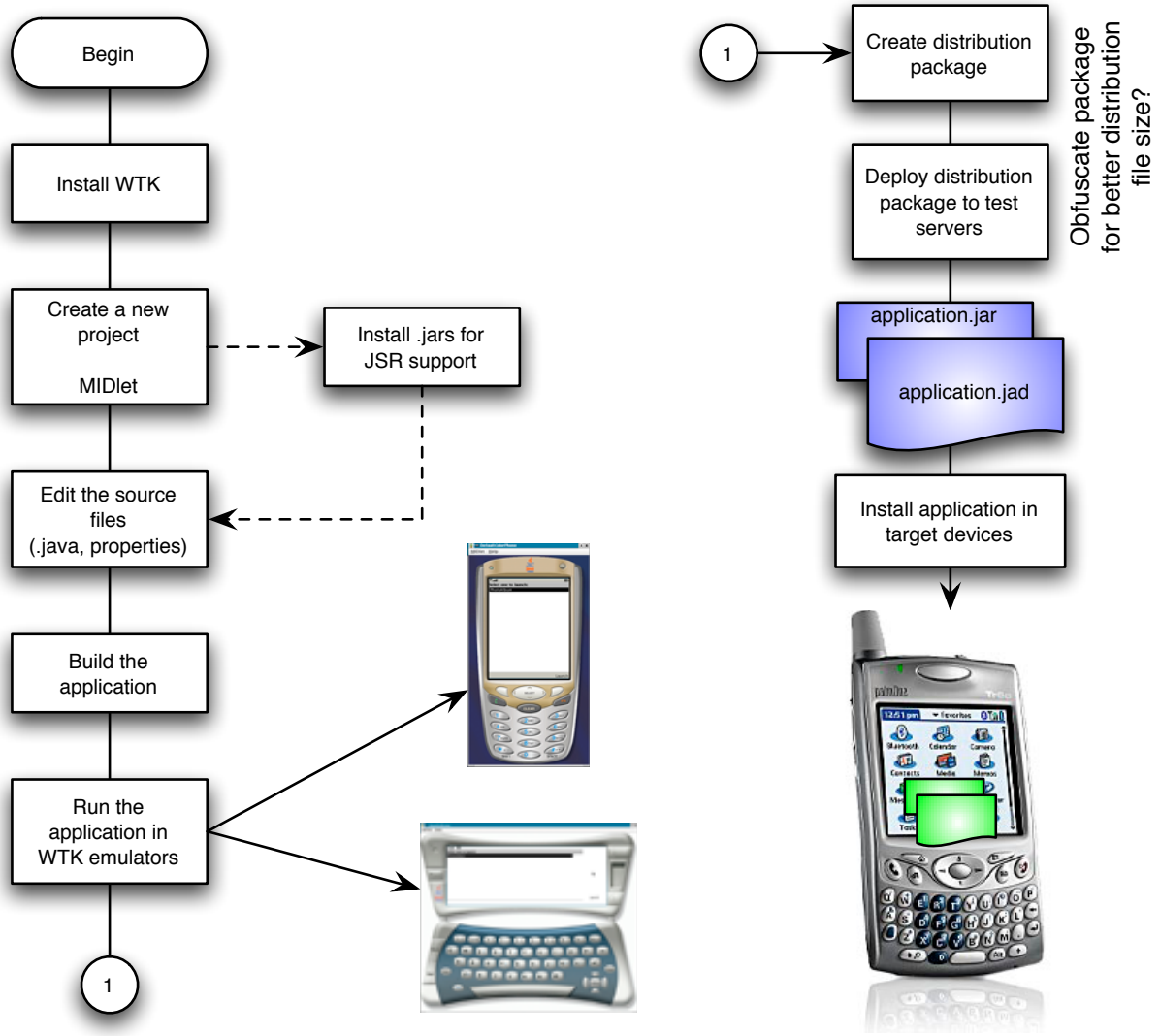
- **Using J2ME provides the widest possible audience**
  - **Carriers may not like this**
  - **Handset issues**
- **Allow the application to be installed in third-party phones?**
  - **Who installs the application? In-store? Over the air?**
- **User interface issues**
  - **Must code to the lowest UI common denominator to support the widest range of platforms**
- **No accurate way to determine device capabilities from J2ME**

Building the Application

Ugh!
Only works with Windows and Linux!

- **Development environment**
  - **J2ME - Wireless Toolkit 2.1 or later**
  - **Java 1.4.2 or Java 5**
  - **Third-party development tools (Netbeans, Eclipse, etc.)**
- **Testing environment**
  - **At least one device in the target category**
- **Additional support:**
  - **Devices must support CLDC 1.1 and MIDP 2.0**
    - **Connected Limited Device Configuration**
    - **Mobile Information Device Profile**
  - **JSR-75: Access to the device's file system**
    - **Also provides access to PIM functionality**
    - **Support for multiple root file systems in the device**

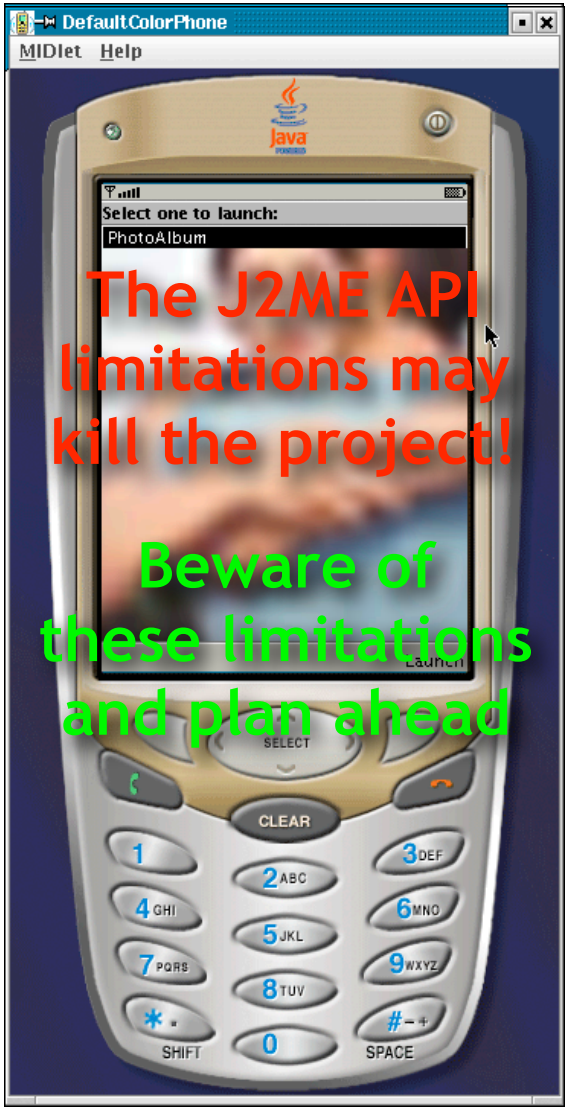
Building the Application



Building the Application (J2ME Limitations)

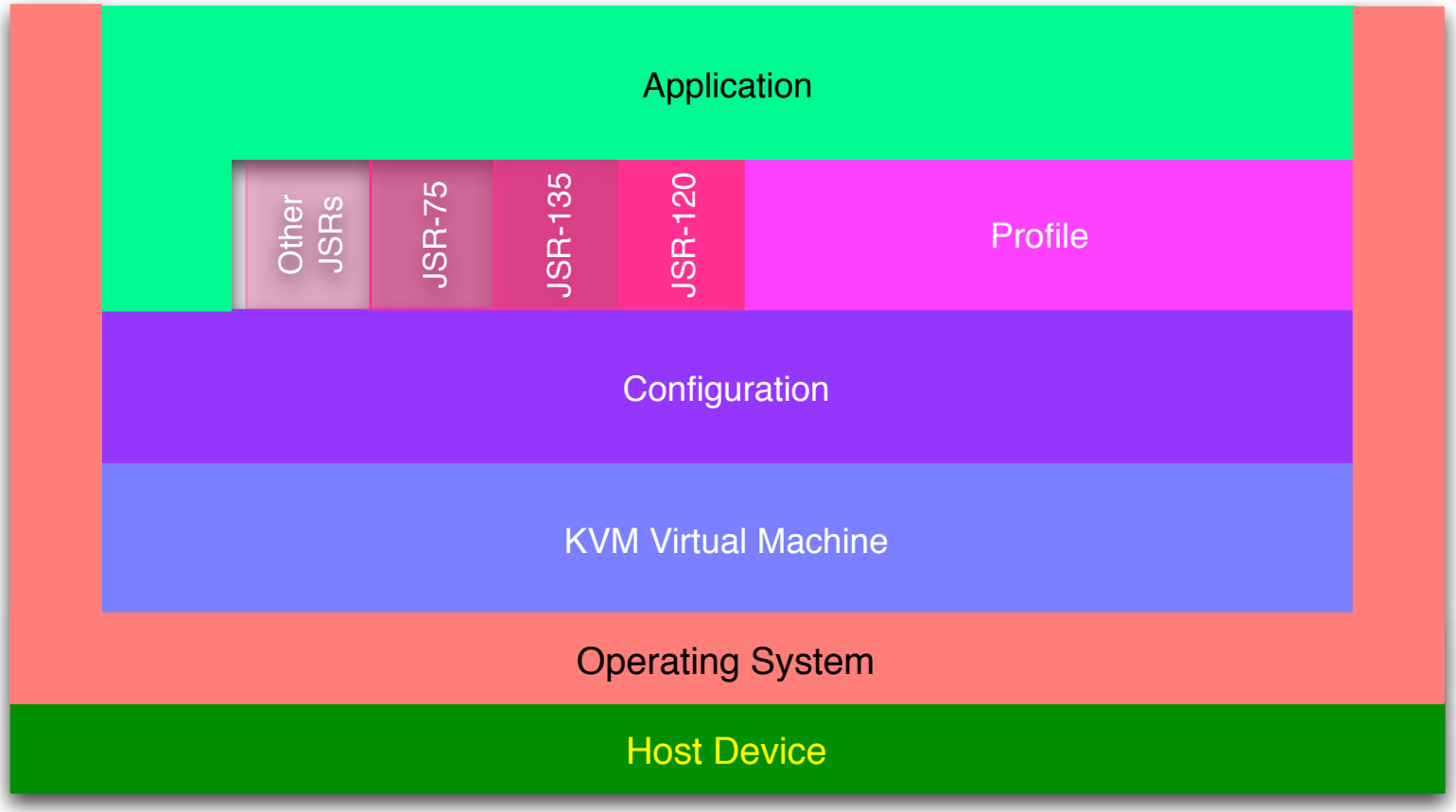
- The user wants to see the contents of the PhotoPak as thumbnails, just like in the phone's photo browser
- Example: J2ME/JSR-75 limitations may prevent this!
  - Only graphics format supported: PNG
  - No raster manipulation, no resizing
  - Limited or uneven access to file system
- The user may only be able to see one photo at a time!
  - File system issue: limit the number of target devices
  - Graphics: use a 3rd-party graphics manipulation library

Building the Application (J2ME Limitations)



- The user wants to see the contents of the PhotoPak as thumbnails, just like in the phone's photo browser
- Example: J2ME/JSR-75 limitations may prevent this!
  - Only graphics format supported: PNG
  - No raster manipulation, no resizing
  - Limited or uneven access to file system
- The user may only be able to see one photo at a time!
  - File system issue: limit the number of target devices
  - Graphics: use a 3rd-party graphics manipulation library
- Big Issue: No JNI operations - J2ME is completely sandboxed!
  - Complete reliance on JSRs and manufacturers' whims

Building the Application (Constraints)



Applications are constrained by the J2ME configuration, the available profile, and the installed JSR-compliant profiles; the J2ME stack is constrained by the operating system.

Building the Application: JSRs

JSR	Description	WTK	Real world
75	Access to PIM and device file system data	Yes	Yes; uneven support by manufacturers
82	Bluetooth integration	Yes	Yes
120	Wireless messaging API	Yes	Some times; late 2006
135	Mobile media API	Yes	Yes; pervasive late 2006
172	Web services API	Yes	Rarely
184	Interactive 3D API	Yes	2007
185	Wireless messaging API	Yes	2007

Most of these APIs (including MIDP and CLDC) have existed since 2001 or so. In real-life applications, the biggest issue is that they aren't evenly supported by manufacturers and some of the JSRs seem to have been written vaguely on purpose to protect vendors' feature sets.

Manufacturer and Carrier Considerations

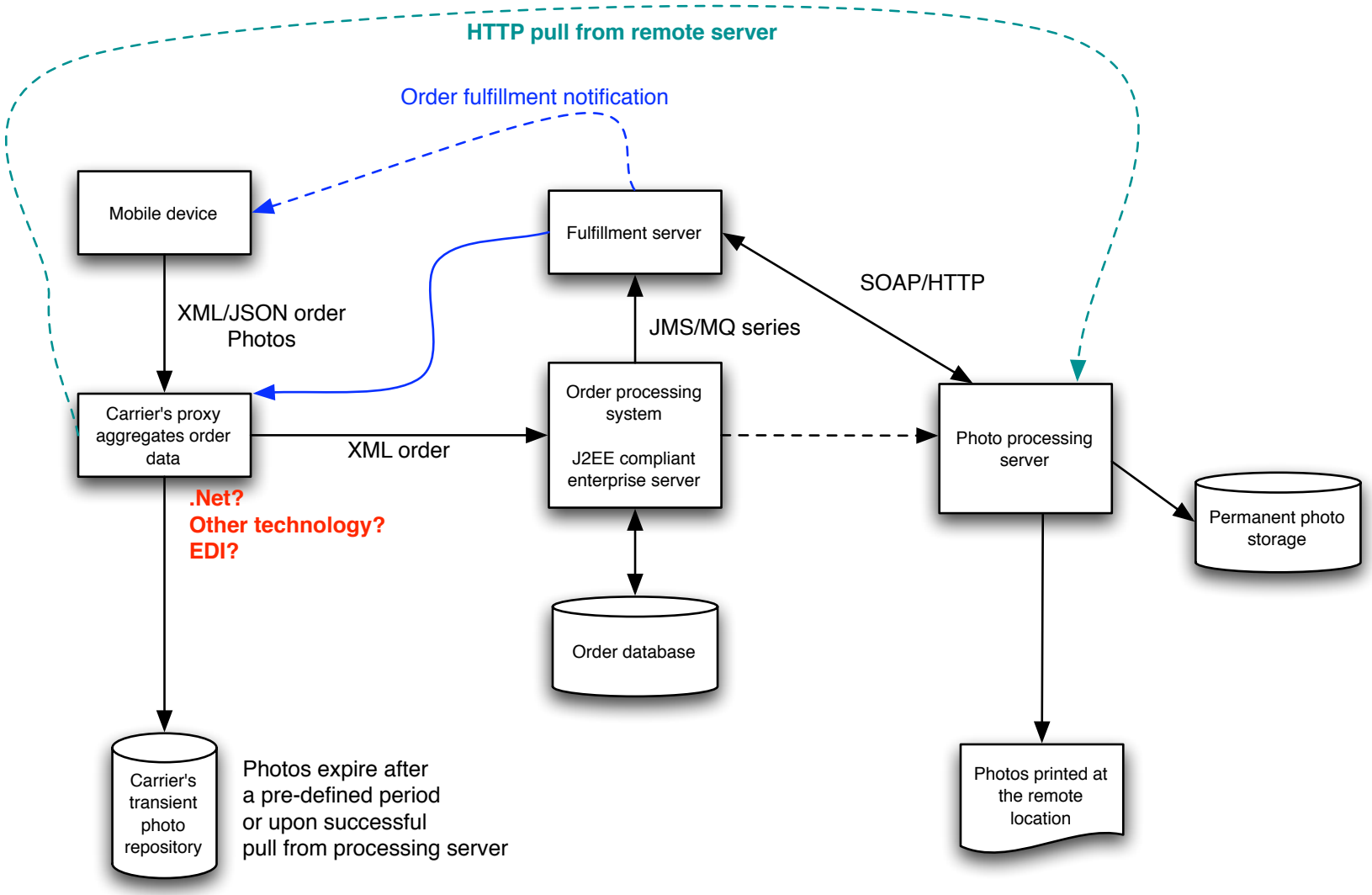
- **70% of all mobile devices are sold by a carrier or franchise**
  - **Device subsidies**
- **Once hooked up with a carrier, it's unlikely that you'll do business with a different carrier**
  - **Picking the right carrier is an art**
  - **Lots of back-and-forth negotiations**
- **Network infrastructure support?**
  - **CDMA: Code Division Multiple Acces**
  - **BREW: Qualcomm's "Son of CDMA"**
  - **GPRS: General Packet Radio Service**

It's a business model, not only a technology!

Application Integration and Deployment

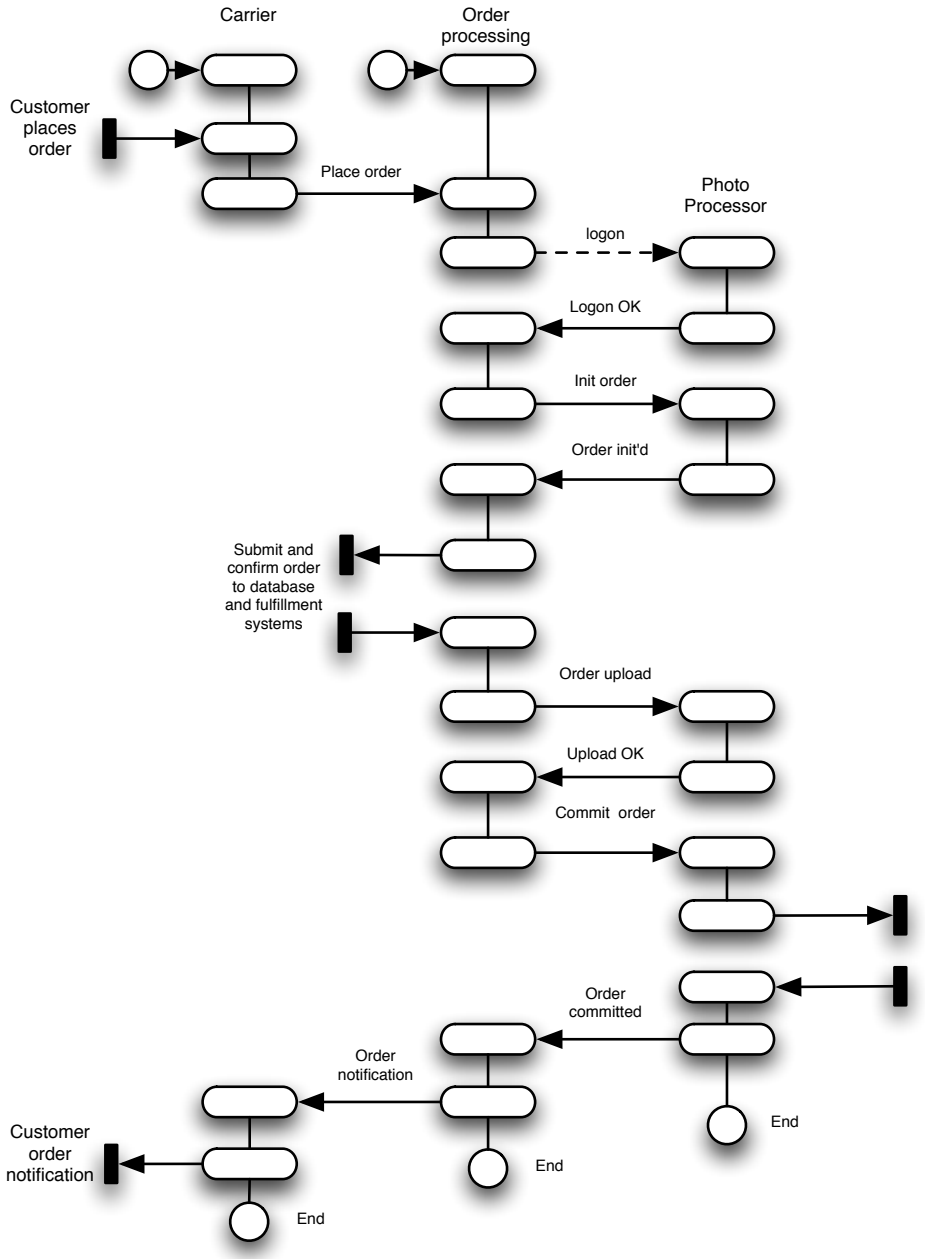
- **The manufacturers and carriers certify the application**
- **The user experience is defined in terms of the device:**
  - **Post a request**
  - **Receive primary acknowledgment from carrier**
  - **Upload data/photos/whatever**
  - **Carrier uses a proxy - best way to re-sync if something goes wrong**
  - **Carrier notifies service provider**
  - **Service provider fetches data and fulfills service**
  - **Service provider notifies carrier OR customer that order is fulfilled**
  - **User receives SMS notification**

Application Integration End-to-End



Application States

- **Many intermediate systems involved in the transaction**
- **Lots of asynchronous operations**
  - **Order is placed after the photos are uploaded to the carrier's intermediate server**
- **Lots of wait states**
- **Reduce the wait at the mobile device as much as possible**
  - **Don't hog the line**
  - **Transfer the minimum amount of data to proxy**



Q&A

Eugene Ciurana
eugenex@walmart.com
Chief Architect, Advanced Technology Group
Walmart.com