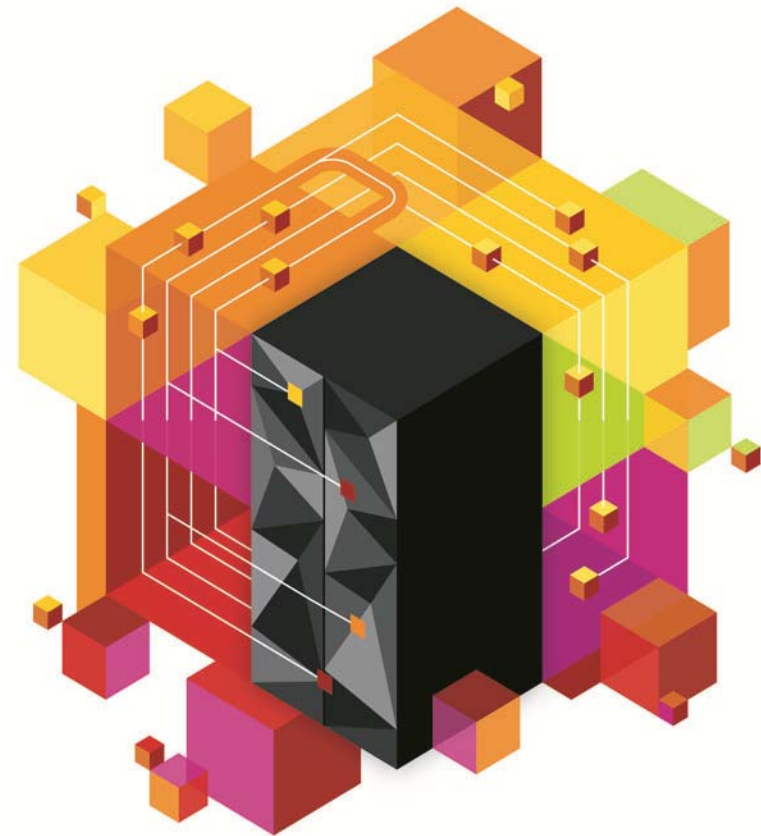




# IBM zEnterprise Technology Summit

Presenter – Title

Date



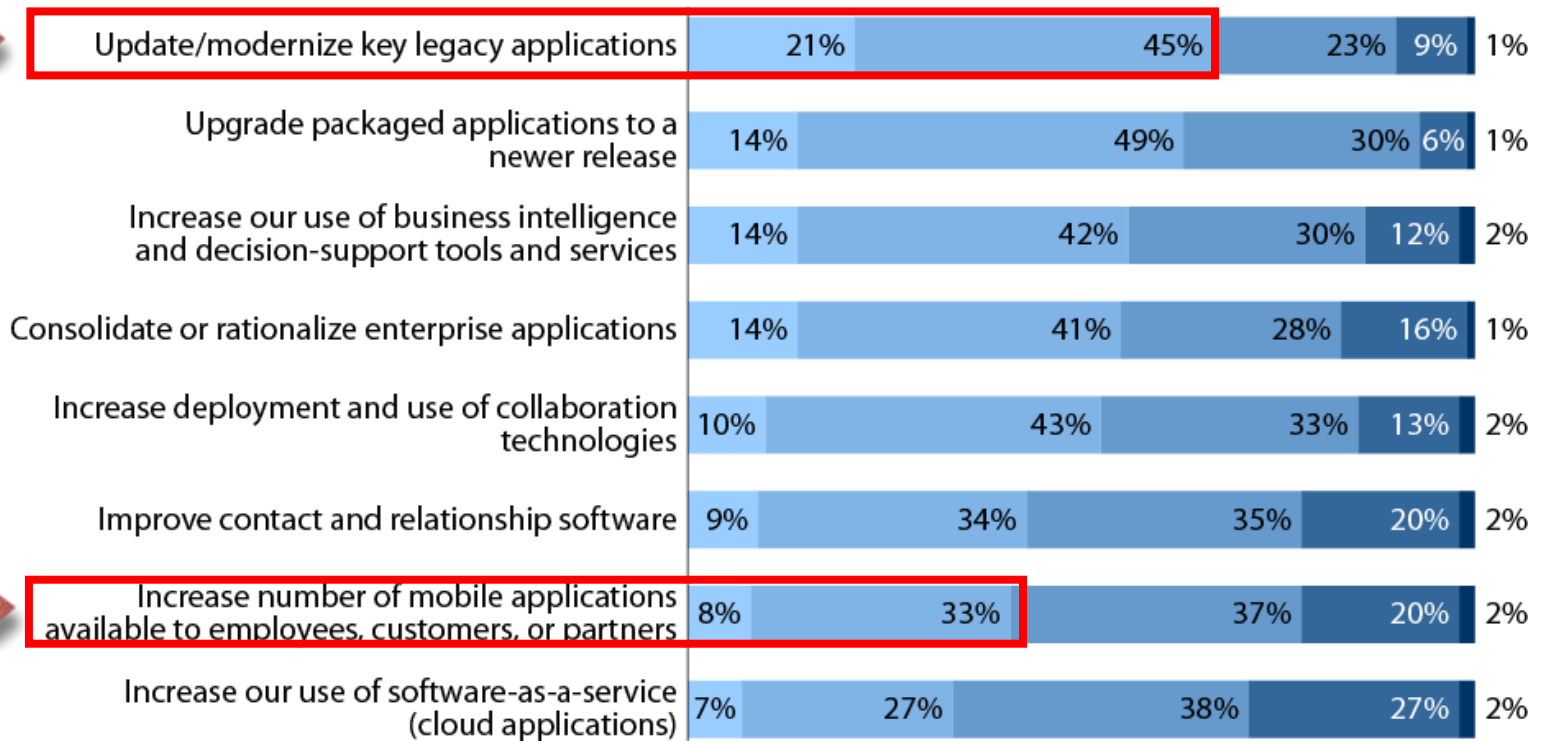


## 6 - Mobile and Cloud Development Agenda

- **Introduction**
- **Existing application challenges**
- **Mobile application challenges**
  - Turning traditional applications from legacy to mobile ready
  - Fragmentation and developing for multiple mobile platforms
  - Design and functional quality are both critical to the success of Challenge 3: Integrating with existing systems
  - Meeting tight time-to-market requirements  
Mobile is pushing traditional delivery approaches to the breaking point
- **Mobile solution overview**
- **Demo: A real world Mobile to CICS Scenario**
- **Q&A**

# Top priority for executives: Updating existing applications

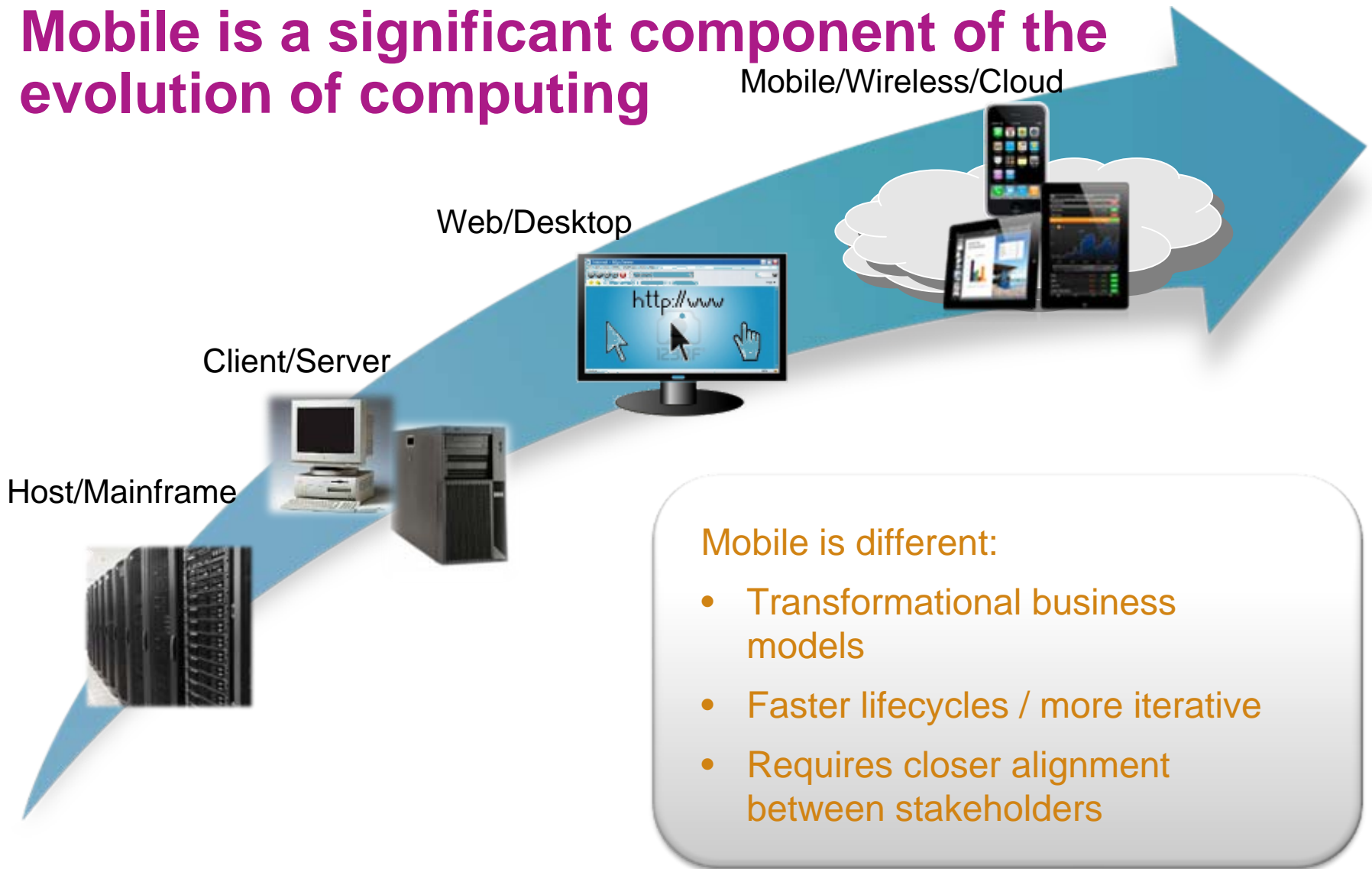
■ Critical priority  
 ■ High priority  
 ■ Low priority  
 ■ Not on our agenda  
 ■ Don't know



Note: Results based on 2,124 software decision makers (percents may not total 100 because of rounding).

Source: Forrsights Software Survey, *Application Retirement — It's Time To Put The Elephant In The Room On A Diet*, February 2011.

# Mobile is a significant component of the evolution of computing



## Mobile is different:

- Transformational business models
- Faster lifecycles / more iterative
- Requires closer alignment between stakeholders

# Mobile is a Mandatory Transformation

## 10 Billion

devices  
by 2020

## 61%

of CIOs put mobile as  
priority

## 45%

increased productivity with  
mobile apps



# Mobile presents an enormous set of opportunities...

## Business to Employee



- Increase worker productivity
- Extend existing applications to mobile workers
- Reducing fuel, gas, or fleet maintenance costs (relevant in particular industries)
- Increase employee responsiveness and decision making speed
- Resolve internal IT issues faster
- Reduce personnel cost (utilizing personal owned instead of corporate issued devices)

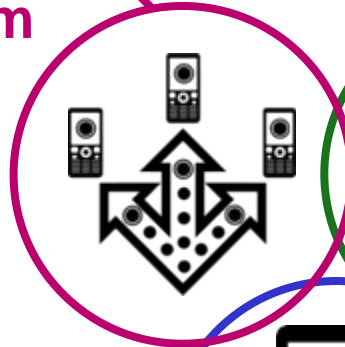
## Business to Consumer



- Improve customer satisfaction and brand perception
- Deeper customer engagement and loyalty
- Drive increased sales through personalized offers
- Make your services available anywhere
- Customer service
- Competitive differentiator
- Deeper insight into customer buying behavior for up sell and cross sell

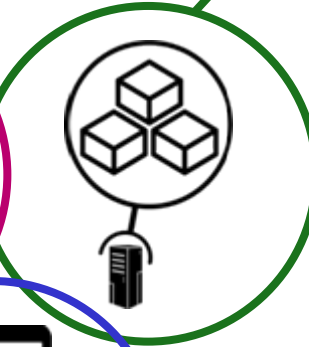
# IBM Comprehensive Mobile Strategy

## Extend & Transform



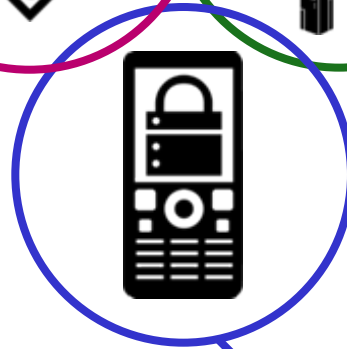
## Build & Connect

*Build* mobile applications  
*Connect* to, and *run*  
 backend systems in support  
 of mobile



## Key Capabilities

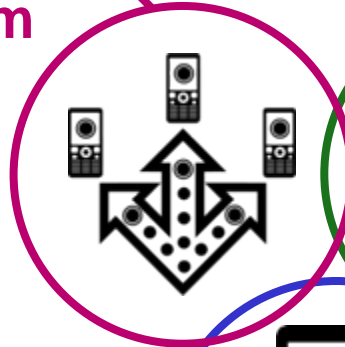
- Multiplatform mobile web, hybrid and native app development
- Enterprise data, service, and application integration
- Iterative teamwork



## Manage & Secure

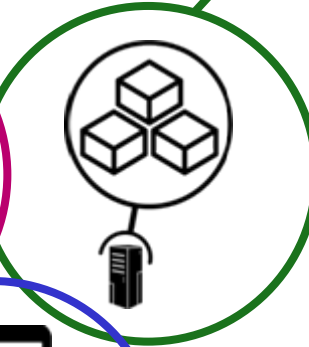
# IBM Comprehensive Mobile Strategy

## Extend & Transform



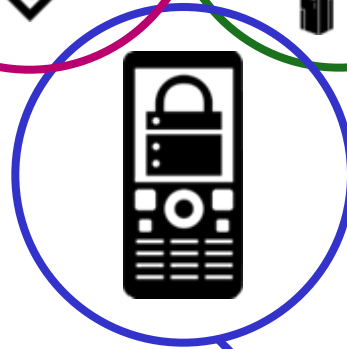
## Build & Connect

*Build* mobile applications  
*Connect* to, and *run* backend systems in support of mobile



## Key Capabilities

- Multiplatform mobile web, hybrid and native app development
- Enterprise data, service, and application integration
- Iterative teamwork



## Manage & Secure

*Manage* mobile devices, services and applications

*Secure* my mobile business

## Key Capabilities

- Mobile Governance
- Device analytics and control
- Secure network communications & management

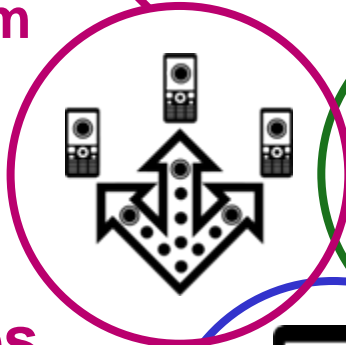


# IBM Comprehensive Mobile Strategy

## Extend & Transform

*Extend* existing business capabilities to mobile devices

*Transform* the business by creating new opportunities



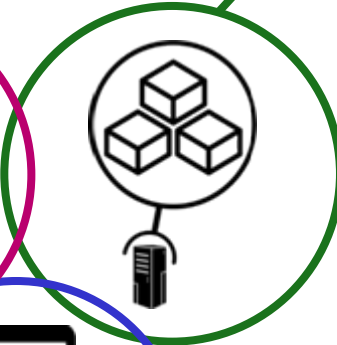
## Key Capabilities

- Strategy, planning and implementation
- Mobile-enabled solutions including analytics, commerce, and social business
- Mobile as a service

## Build & Connect

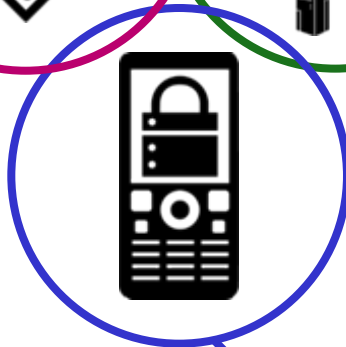
*Build* mobile applications

*Connect to, and run* backend systems in support of mobile



## Key Capabilities

- Multiplatform mobile web, hybrid and native app development
- Enterprise data, service, and application integration
- Iterative teamwork



## Manage & Secure

*Manage* mobile devices, services and applications

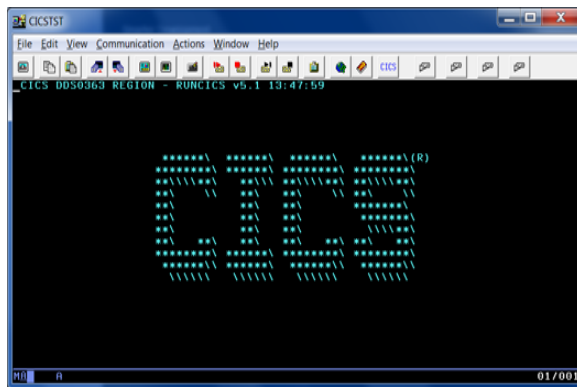
*Secure* my mobile business

## Key Capabilities

- Mobile Governance
- Device analytics and control
- Secure network communications & management

# Existing applications

Our focus is to help modernize existing z Enterprise CICS application architectures by extending into a mobile client platform with strategies requiring little to no modifications required to the backend applications.



# Architectures: CICS

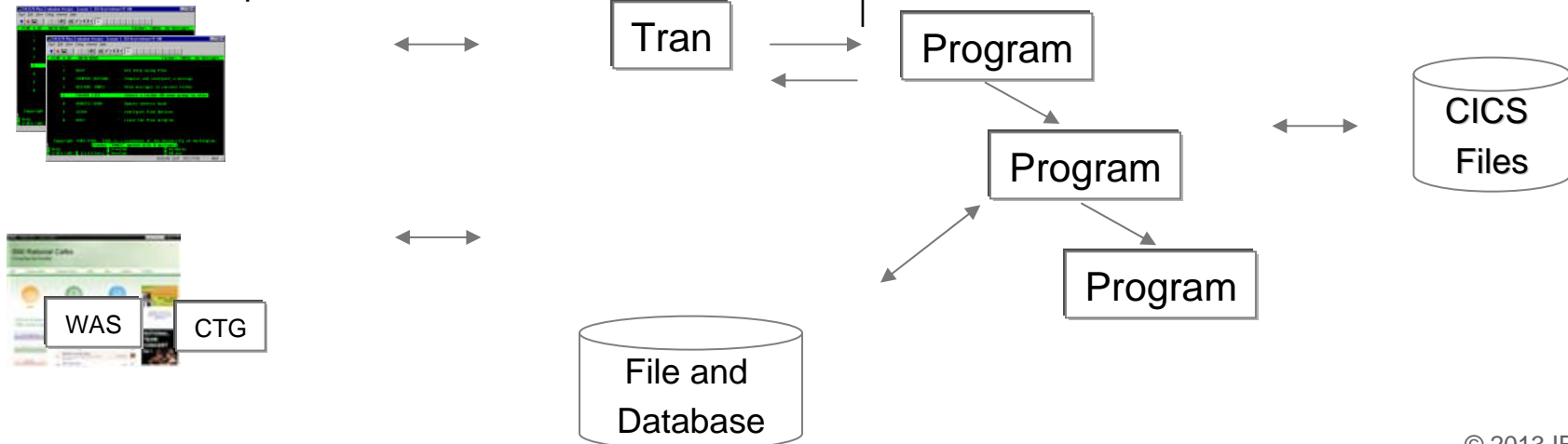
## Modernization Requirements

- Make business logic more accessible as services
- Make Data more accessible
- Integrate with new Java processes
- Link with process and rules engines
- Don't impact my current transactional throughput

## Modernization Strategies

- Modularize (and service enable) code
- Migrate legacy data stores to relational DB2
- Modify data access / add data layer to access relational DB2
- Rewrite and/or migrate code to rules and/or WAS

3270 - BMS Maps



# Architectures: IMS

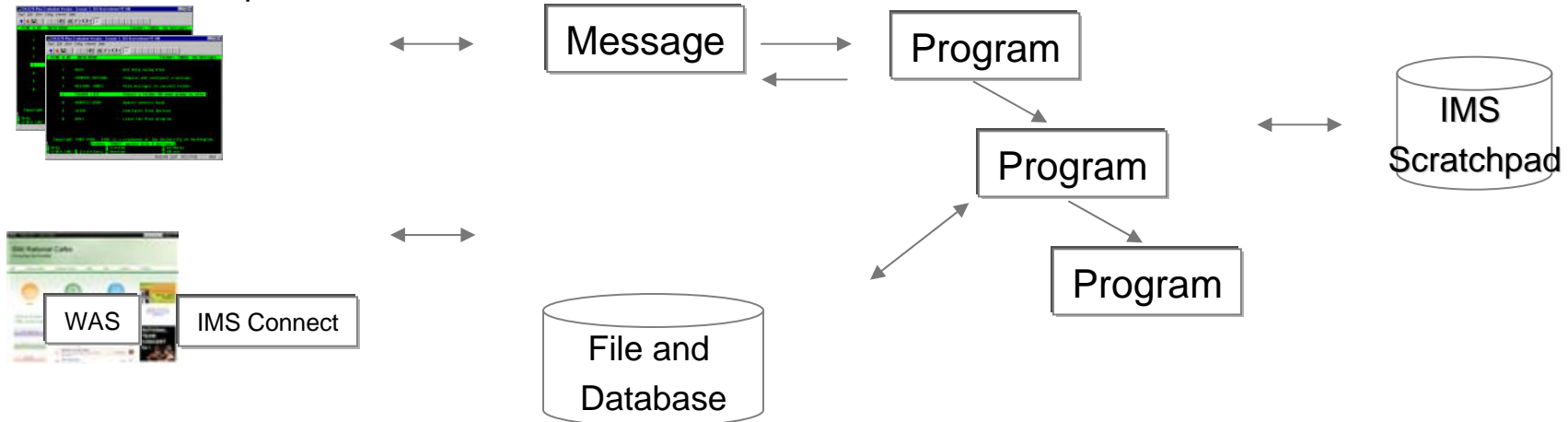
## Modernization Requirements

- Make business logic more accessible as services
- Make Data more accessible
- Integrate with new Java processes
- Link with process and rules engines
- Don't impact my current transactional throughput

## Modernization Strategies

- Modularize (and service enable) code
- Migrate legacy data stores to relational DB2
- Modify data access / add data layer to access relational DB2
- Rewrite and/or migrate code to rules and/or WAS

### 3270 – MFS Maps



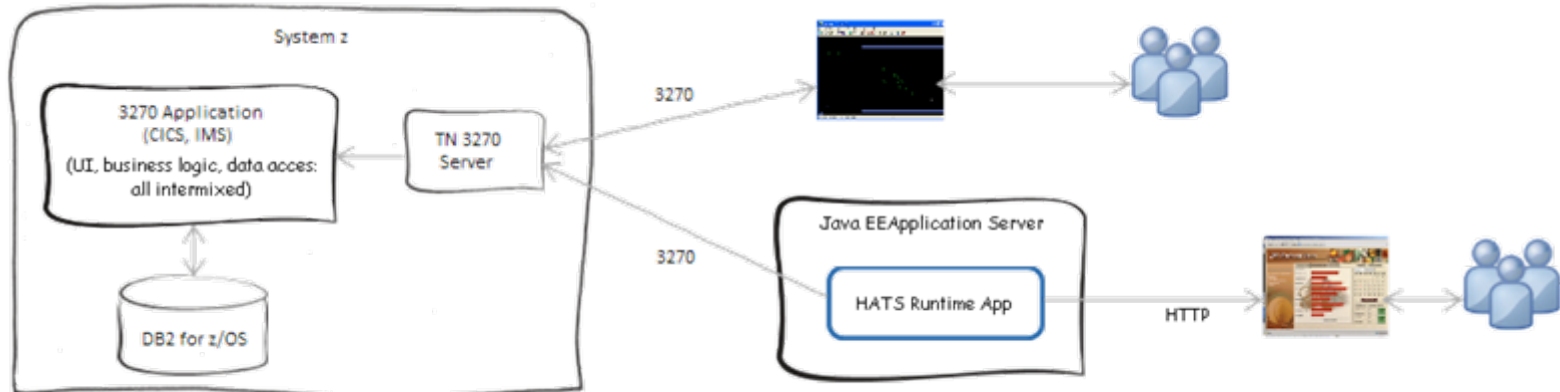
# Strategies: Re-facing

## Requirements

- Provide web/mobile access to existing apps fast
- Reduce IT operational costs associated with maintaining multiple emulators/versions across desktops
- Applications cannot be modified (lack of source code, lack of skills, too risky, etc)
- Continue providing access via 3270 for power or back-office users

## Solution Overview

- Eclipse-based tool for developing and testing
- WebSphere/JEE or Eclipse/Notes runtime
- Quickly transforms green screen applications to Web, Mobile, Portal and rich client interfaces
- Retain 3270 look/feel/navigation if needed
- Iteratively improve application look and usability
- No changes required to existing applications



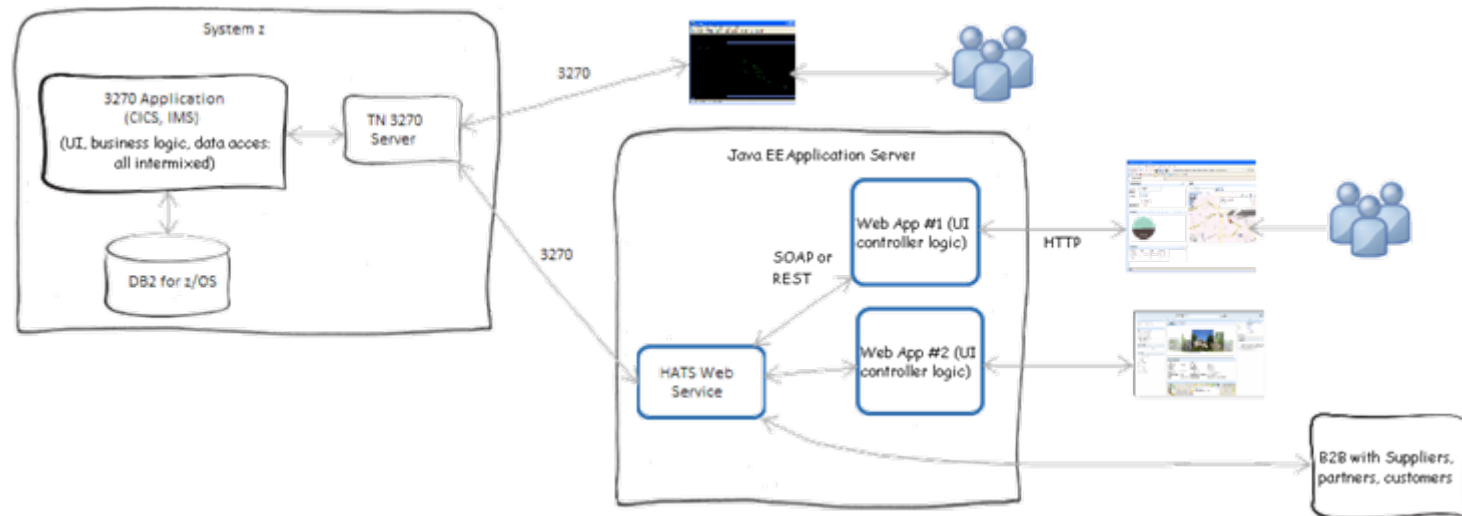
# Strategies: Service Wrapping

## Requirements

- Quickly expose existing business logic for integration with other systems
- Web UI needs to drive application flow and support integration with other services (mashup scenario)
- Applications cannot be modified (lack of source code, lack of skills, too risky, etc)
- Keep access via 3270 intact

## Solution Overview

- Web service-enabled flows through 3270 applications without changing the existing applications
- Enables integration of host business processes and data with other Web, Portal, Mobile, and rich client applications



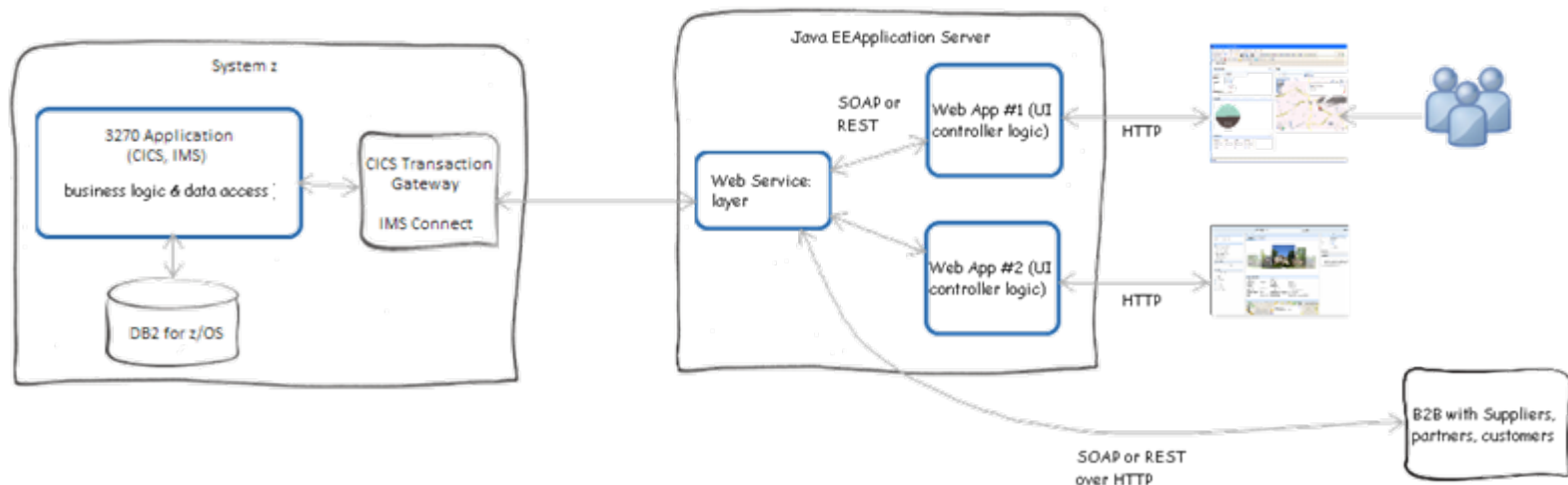
# Strategies: Application Refactoring

## Requirements

- Meet new business needs that overcome the constraints of the old system
- More flexibility and improved ability to respond quickly to business requirements for a competitive advantage
- Enable business process automation by integrating systems (internal, suppliers, partners, clients)

## Solution Overview

- “De-tangle” UI from COBOL business logic and create programs & service programs that have well-defined interfaces
- Build a services layer that exposes these reusable programs as web services, either with Java or EGL wrappers
  - ▶ Driven by new web/mobile applications
- Complement RD for z tools with other Rational tools for application understanding and analysis



# Key mobile development and delivery challenges

## Fragmentation and developing for multiple mobile platforms

- Highly fragmented set of devices, platforms, languages, and tools complicates development, test, and operations



## Delivering high quality apps

- Consumers demand a high quality user experience where quality is influenced as much by design as it is by function



## Accelerated time to market requirements

- Higher frequency of new releases puts added pressure on teams to deliver on time and with high quality



## Connecting apps with existing enterprise systems

- Apps typically need to leverage existing enterprise services, which must be made mobile-consumable
- Enterprise wireless networks are running out of bandwidth to accommodate employee devices





# Challenge : Fragmentation and developing for multiple mobile platforms

- **Several major platforms with their own**
  - Tools
  - Languages, APIs, and programming models
  - App stores
  - Ecosystems
- **Fragmentation within platforms, including**
  - Physical device differences
  - Version incompatibilities
  - Vendor customizations
- **And this market is moving very fast – new things are introduced all the time**
- **Worklight and RDzEnterprise addresses this challenge by supporting an open, hybrid mobile application architecture**



# Challenge: Design and functional quality are both critical to the success of a mobile application

- **Customers now know better – they know what good user experience is and they demand it**
- **Business-to-Consumer (B2C) applications**
  - Line of Business wants these applications to improve customer satisfaction, drive customer engagement and loyalty
  - These apps are the new faces of the business
  - User experience is key to brand perception
- **Business-to-Employee (B2E) applications**
  - Increase worker productivity, improve responsiveness and decision making speed
  - Poor user experience directly compromises the value of the application



## Customer Ratings





# Perceived app quality is influenced as much by design quality as it is by functional quality

## Design Quality

- User experience is critical for mobile applications
- Mobile applications typically require you to rethink how your customers interact with your business
- Line of business expects these applications to improve customer satisfaction, drive engagements, and loyalty

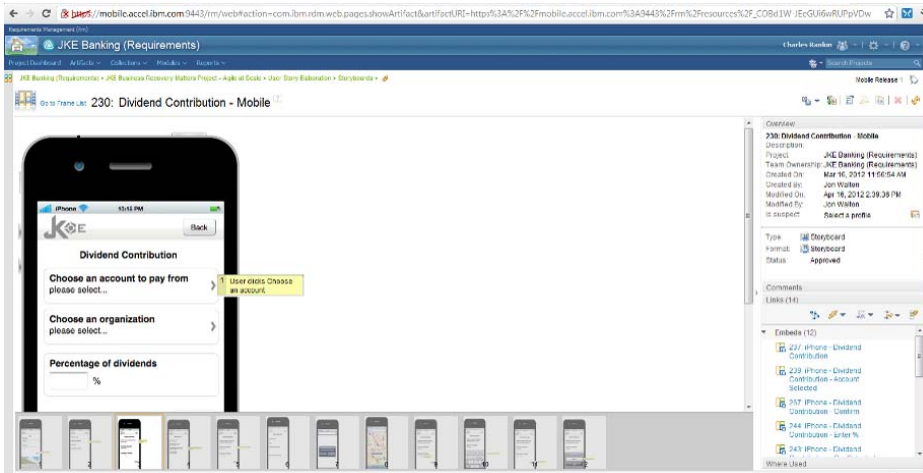
## Functional Quality

- Planning tests against all combinations of devices, Oses, carriers in fragmented market yields exponential number of test cases
- Testing is complicated by unconventional ways of interacting with mobile devices (camera, accelerometer, gestures, speech)
- Maintaining large library of devices in-house is cost prohibitive

**Goal:** deliver apps that align with business goals and are perceived as high quality – both from a user experience and functional point of view

# Ensuring high app quality – both design and functional

## *Integrated requirements, planning, and quality management with Rational Requirements Composer and Rational Quality Manager*



Mobile Device Cloud

### Client Challenge

Delivering apps that align with business goals and are perceived as high quality – both from a user experience and functional point of view.

### Key Capabilities

- End-to-end integration across the development lifecycle, including design and test phases
- UI sketching and storyboarding
- Design requirements management
- Collaborative reviews and approvals
- Test plan optimization to minimize number of tests required to be executed
- Integration with mobile “Device-cloud” testing services (e.g. DeviceAnywhere and Perfecto Mobile) and automated testing

# Challenge: Meeting tight time-to-market requirements

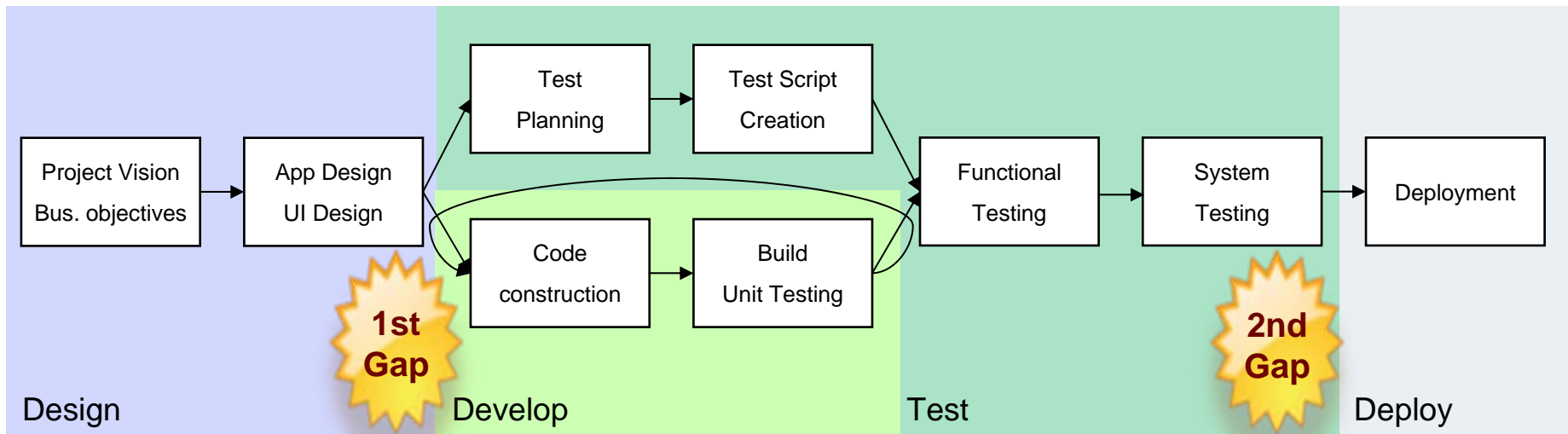
## Mobile is pushing traditional delivery approaches to the breaking point



	Mobile Apps	Desktop Apps
<b>Time-to-market</b>	Weeks to Months	Months to Years
<b>Frequency of updates</b>	Once every several weeks	12-18 month cycles

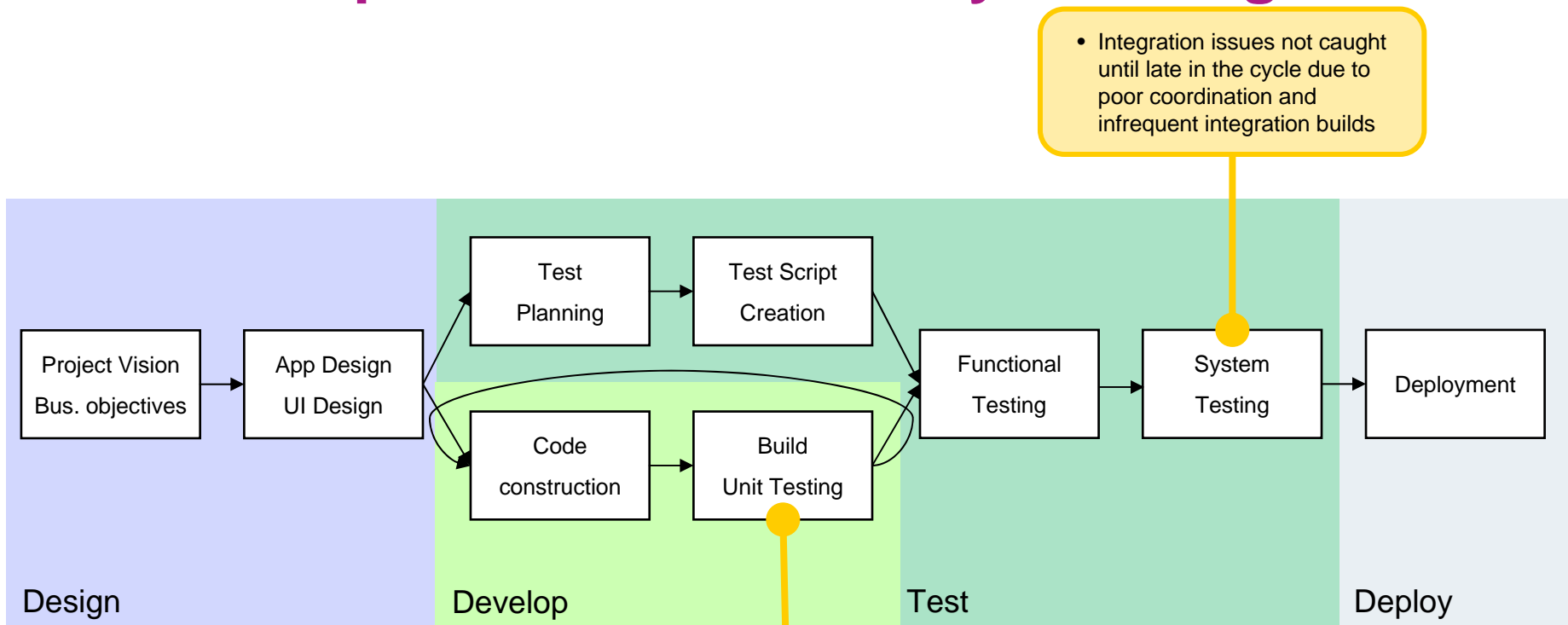


# Factors affecting project velocity



- **There are 2 key factors affecting project velocity:**
- **Gap 1: amongst Line of Business, Development and Test teams (ALM)**
  - Late rework due to misalignment of stakeholders
  - Slow progress due to hand-off errors and delays between team roles
- **Gap 2: between Development/Test and Operations Team (DevOps)**
  - Slow cycle/iteration times due to DevOps challenges

# Mobile unique build and delivery challenges



- Mobile platforms have unique SDKs for building and packaging apps
  - Builds must be spread across multiple environments (e.g. Mac OS, Windows, Linux, etc)
  - Impractical for developers to maintain SDKs on their workstations, which makes it difficult to test and catch potential build issues
- Managing security certificates and ensuring the app gets built with the right set
- Dealing with platform-specific constraints on the number of devices that an app can be tested on

# Challenge: Integrating with existing systems

- **Mobile applications need to connect to enterprise back-end data and services**
  - Existing programs and services may need to be refactored to be made mobile-consumable
  - Mobile-optimized services have different characteristics than traditional web services (payload size, incremental data access, etc)
- **Mobile application development lifecycle needs to bridge the multiple teams responsible for different parts of the mobile application**
- **Testing multi-tiered mobile applications can be slowed due to:**
  - All integrations with back-end must be available to test entire app through the UI
  - Test environments are expensive, difficult and time consuming to configure
  - Difficult and time consuming to isolate defect root cause
  - Agile methodologies need fast iterations but testing delays are becoming a bottleneck





# Solution: Hybrid Mobile

## Web

- HTML, JavaScript, CSS
- Accessed from a mobile web browser
  - No device-specific capabilities



## Mobile Web

- HTML, JavaScript, CSS
- Accessed from a mobile web browser; mobile-optimized UI
- Limited access to lower-level device capabilities



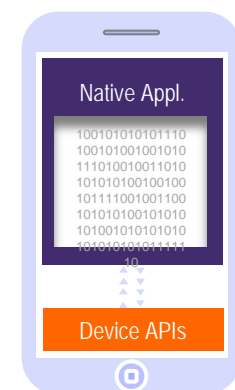
## Hybrid Mobile

- HTML, JavaScript, CSS, with optional native code
- Installed and run like a native mobile app; mobile-optimized UI
- Access to lower-level device capabilities



## Native

- Native code
- Access to full set of lower-level device capabilities



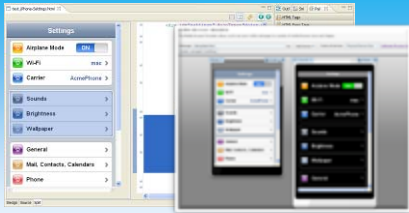
*Richness of Mobile Presentation / Services*

*Portability (cross-device reuse)*

*Maintenance Cost (TCO)*

# Aligning teams across the development lifecycle

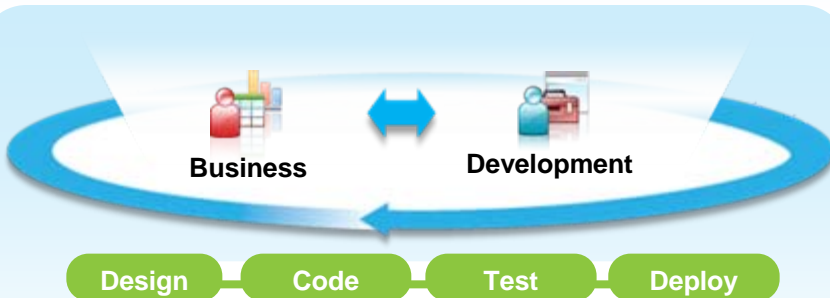
## *Collaborative Lifecycle Management, Rational IDEs, and Worklight*



Construct, debug, and test mobile UIs



Refactor and extend existing logic as mobile-consumable services



*Traceability across the entire development lifecycle*

Open Lifecycle Integration Platform



*End-to-end Lifecycle Management for Mobile Application Development*

### Client Challenge

Mobile apps are typically multi-tiered and require collaboration between multiple teams, including teams responsible for design, development, test, and deployment

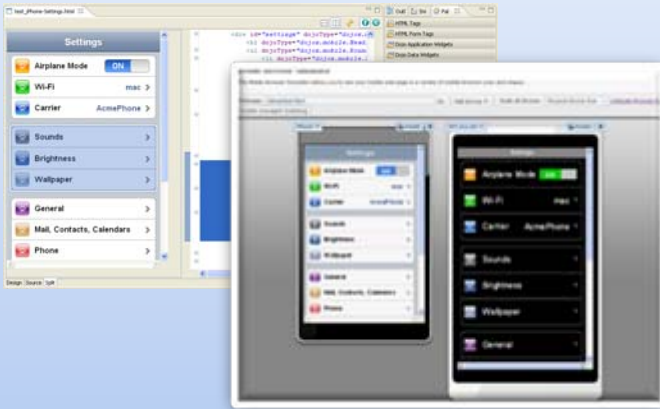
### Key Capabilities

- Common, integrated tool set across all phases of development and components of the mobile solution
- Integration with Worklight Studio to ensure developers have access to plans, tasks, builds, and code from within their development IDE
- Traceability across the entire mobile application development lifecycle – all teams are aware of changes (for example, a changing requirement)

# Rational Developer for zEnterprise 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*

## Integrated multi-platform development environments



**Construct, debug, and test mobile UIs**



**Refactor and extend existing logic on enterprise platforms (System z, Power) as mobile-consumable services**

IBM Worklight 5.0 is now 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

# Accelerating build and deployment time

## Orchestrating native mobile app builds with Rational Team Concert

### Concert



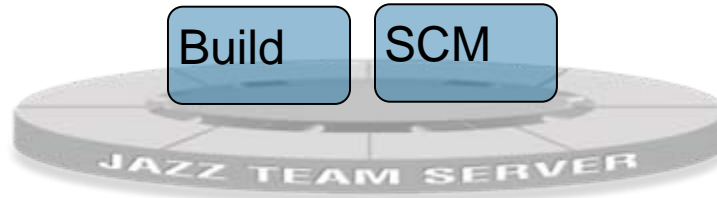
Developer



Mobile devices & emulators



*RTC build engine and Worklight mobile build utility provide a controlled build environment for mobile apps – both native and hybrid*



Team repository of apps



App stores

Mobile build server farm

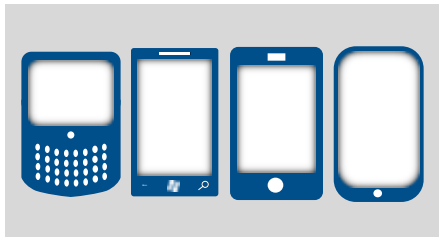
# Worklight Components



**Worklight Studio**  
 The most complete, extensible development environment with maximum code reuse and per-device optimization



**Worklight Server**  
 Mobile middleware offering unified push notifications, version management, security and integration

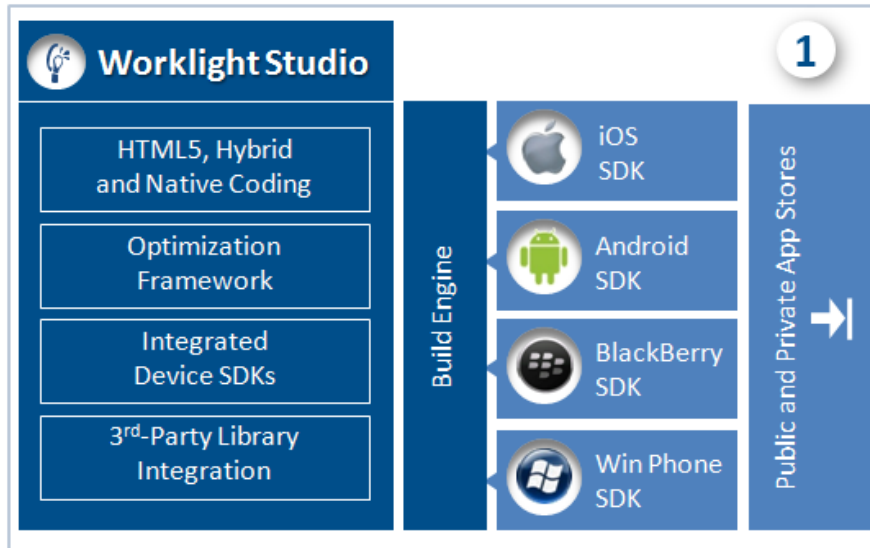


**Worklight Runtime Components**  
 Extensive libraries and client APIs that expose and interface with native device functionality and the Worklight Server



**Worklight Console**  
 A web-based console for real-time analytics and control of your mobile apps and infrastructure

# Worklight Components: Worklight Studio



- Eclipse based, most complete, extensible Mobile IDE
- Build rich mobile application using open standard technologies and commonly available skills
- Single shared code base for all multiple platforms - cross-platform compatibility
- Compatible with:
  - Sencha Touch
  - JQuery
  - Dojo
  - PhoneGap
- Device SDK integration to allow direct
- Tooling for back-end connectivity utilities (Worklight Adapters)

# Studio- WYSIWYG Development Environment

The screenshot displays an IDE interface for developing an Android application. The main workspace shows a WYSIWYG view of a mobile application titled "CICS Catalog Manager". The application interface features a dark background with a light blue header and three main buttons: "Browse Catalog", "Inquire an Item", and "Shopping Cart".

The left sidebar shows the Project Explorer, displaying a tree view of the application's structure. The tree view includes folders for "MobileEnterprise", "apps", and "EGUI". The "EGUI" folder is expanded, showing sub-folders for "android", "common", "css", "images", and "js".

The bottom pane shows the source code for the application, which is an HTML file named "EGUI.html". The code includes a DOCTYPE declaration, a license notice, and various meta tags and links for the application.

```

<!DOCTYPE html>

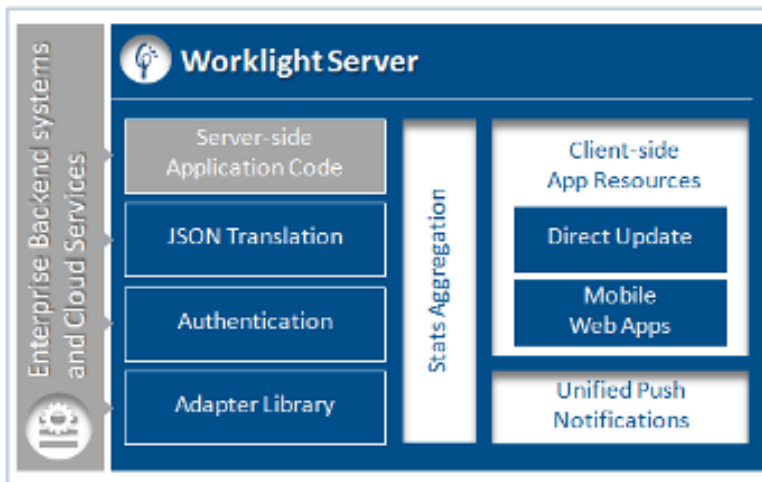
<!-- Licensed Materials - Property of IBM
5725-G92 (C) Copyright IBM Corp. 2011, 2012. All Rights Reserved.
US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with IBM Corp. -->

<html>
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0, user-scalable=0" />
    <title>EGUI</title>
    <link rel="shortcut icon" href="images/favicon.png" />
    <link rel="apple-touch-icon" href="images/apple-touch-icon.png" />
    <link rel="stylesheet" href="css/reset.css" />
  
```

# Worklight Components: Worklight Server

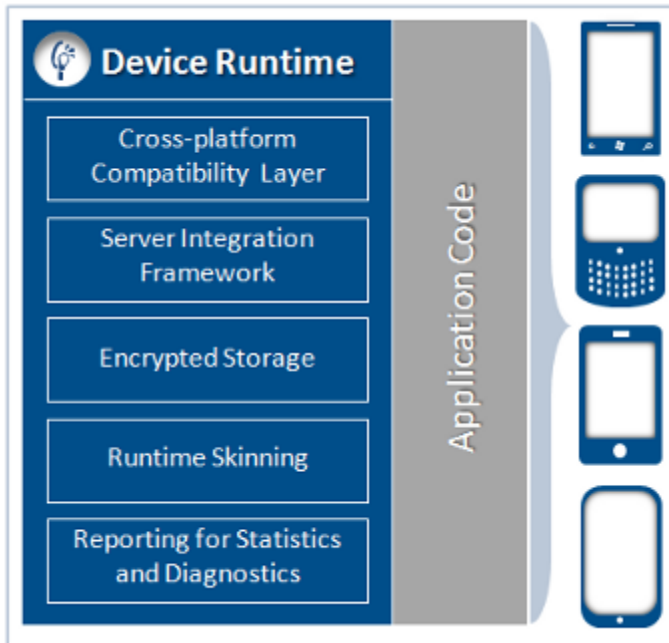
A middleware component which delivers unified notifications, version management, security features, and integration services

- Distribution of mobile web apps
- Enterprise connectivity:
  - Secure client/server connectivity
  - Direct access to enterprise back-end data and transaction capabilities
  - Authentication enforcement
- Client control:
  - Application version management and remote disabling
  - Direct update of application code
- Unified Push Notifications
- Aggregation of usage statistics





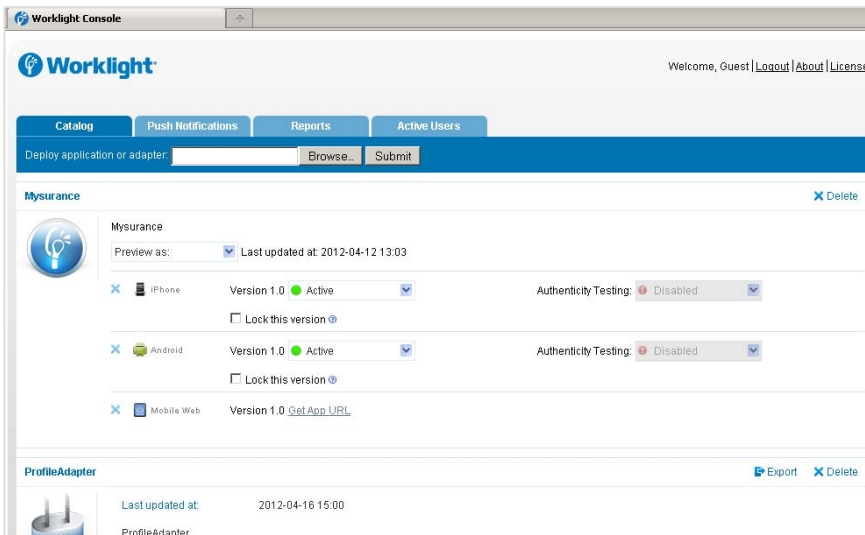
# Worklight Components: Worklight Device Runtime



Extensive libraries and client APIs that expose and interface with native device functionality and the Worklight server

- Cross-Platform Compatibility Layer
- Server Integration Framework
- Encrypted Storage
- Offline Authentication
- Runtime Skinning
- Statistics and Diagnostics Reporting

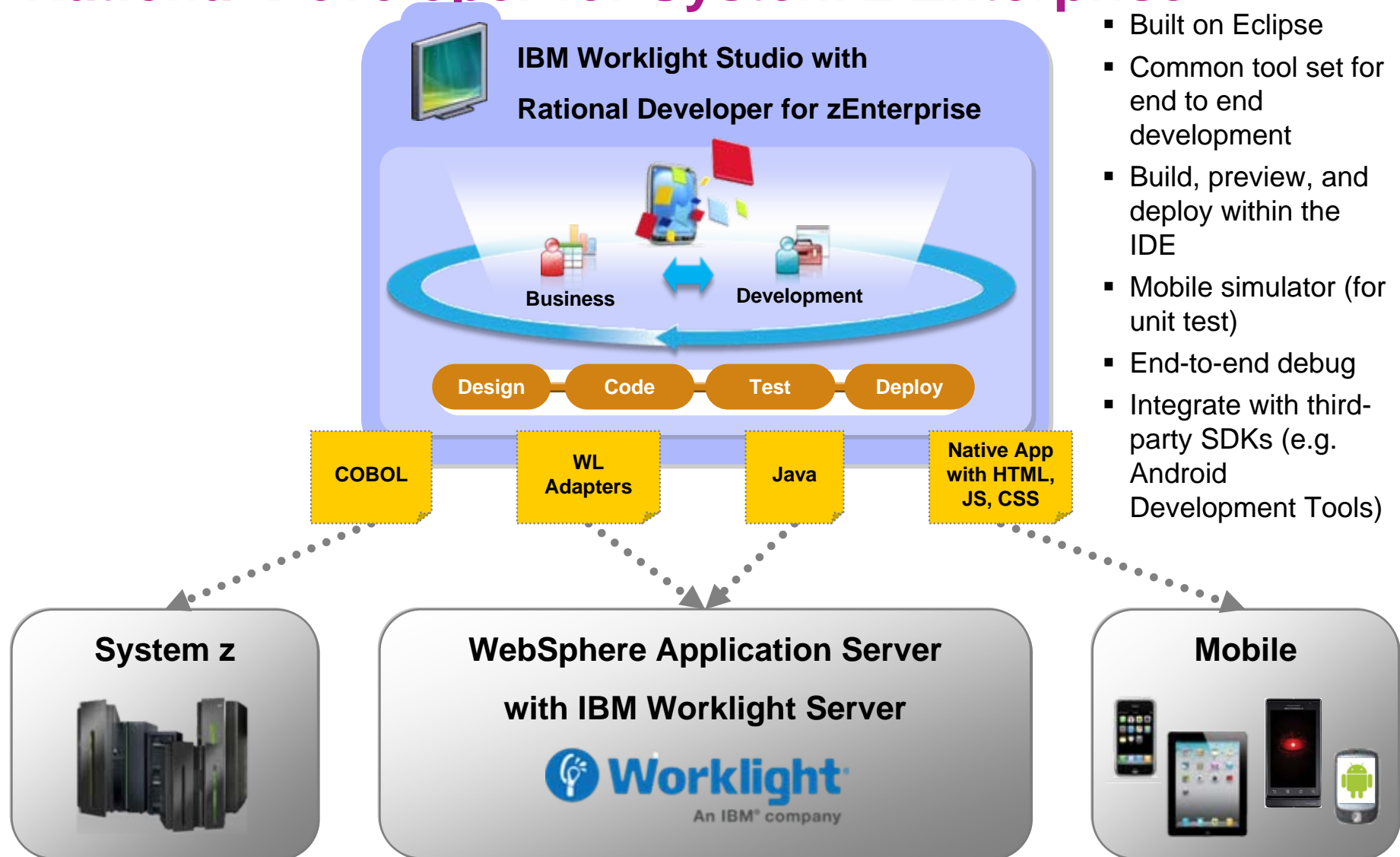
# Worklight Components: Worklight Console



A web-based console for real-time analytics and control of mobile apps and infrastructure

- Application Version Management
- Remote disable
- Push management
- Usage reports
- Reports of custom application events
- Configurable audit log
- Administrative dashboards for:
  - Deployed applications
  - Installed adapters
  - Push notifications

# Rational Developer for System z Enterprise



- Built on Eclipse
- Common tool set for end to end development
- Build, preview, and deploy within the IDE
- Mobile simulator (for unit test)
- End-to-end debug
- Integrate with third-party SDKs (e.g. Android Development Tools)

# CICS Application Walkthrough

CICS main menu screen showing a grid of asterisks and the text 'CICS 0050363 REGION - R08CICS v5.1 13:47:59'.

CICS Catalog Manager application screen showing a list of items with columns for Item, Description, Cost, and Order.

Item	Description	Cost	Order
0010	Ball Pens Black 24pk	2.90	-
0020	Ball Pens Blue 24pk	2.90	-
0030	Ball Pens Red 24pk	2.90	-
0040	Ball Pens Green 24pk	2.90	-
0050	Pencil with eraser 12pk	1.78	Z
0060	Highlighters Assorted 5pk	8.99	-
0070	Laser Paper 28-lb 108 Bright 500/ream	7.44	-
0080	Laser Paper 28-lb 108 Bright 2500/case	33.54	-
0090	Blue Laser Paper 20lb 500/ream	5.35	-
0100	Green Laser Paper 20lb 500/ream	5.35	-
0110	IBM Network Printer 24 - Toner cart	168.56	-
0120	Standard Diary: Week to view B 1/4x5 3/4	25.98	-
0130	Hall Planner: Erasable 36x24	18.85	-
0140	70 Sheet: Hard Back wire bound notepad	5.99	-
0150	Sticky Notes 3x3 Assorted Colors 5pk	5.35	-

## ■ CICS Catalog Manager Application

- COBOL based CICS application
- VSAM File
- Separate Presentation Logic and Business Logic
- Callable Interface - Commarea

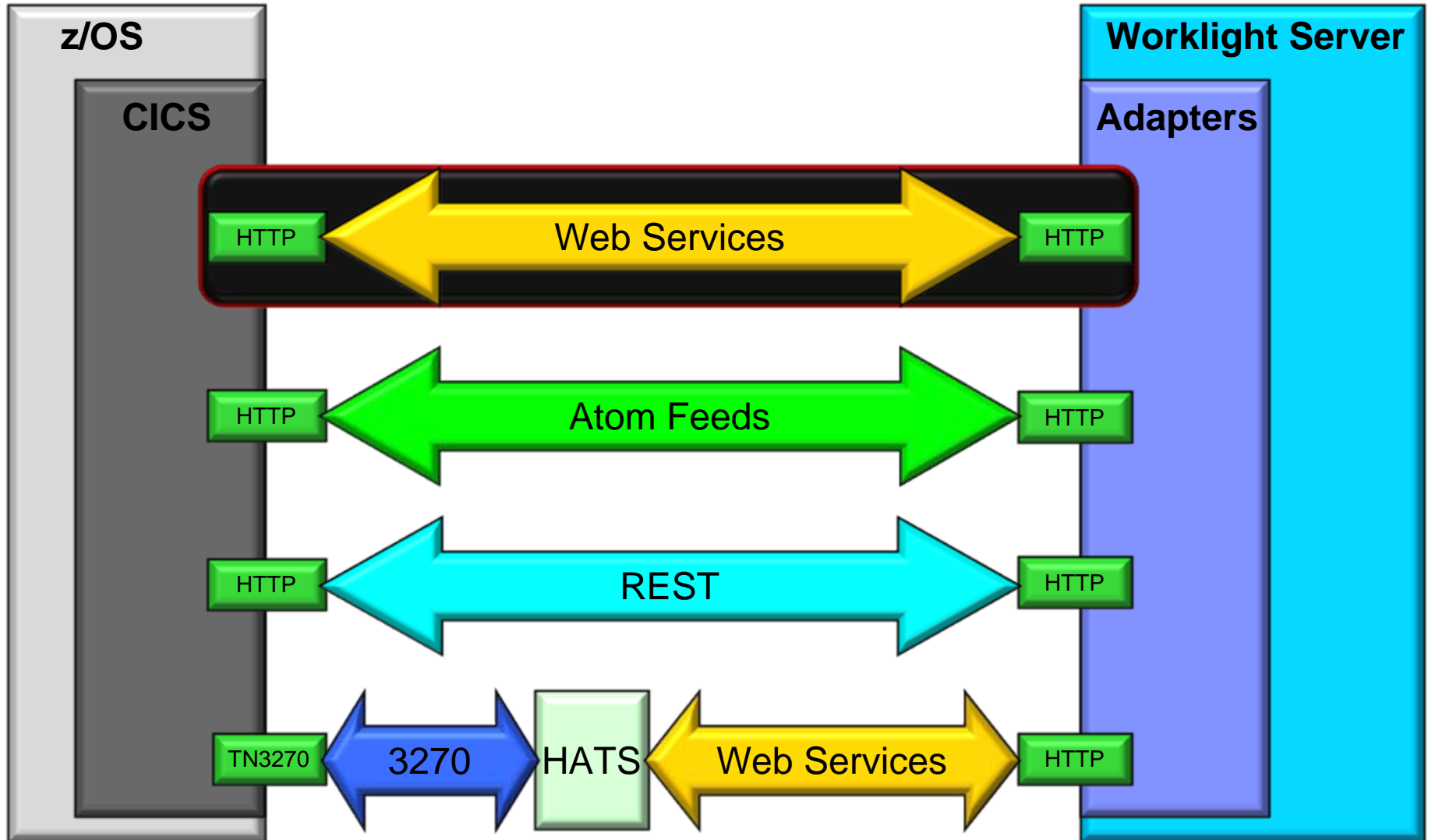
CICS Main Menu screen showing a list of actions: 1. List Items, 2. Order Item Number, 3. Exit.

CICS Catalog Manager application screen showing order details for item 0050 (Pencil with eraser 12pk) with a quantity of 3, user name George, and charge department CICSUSRS.

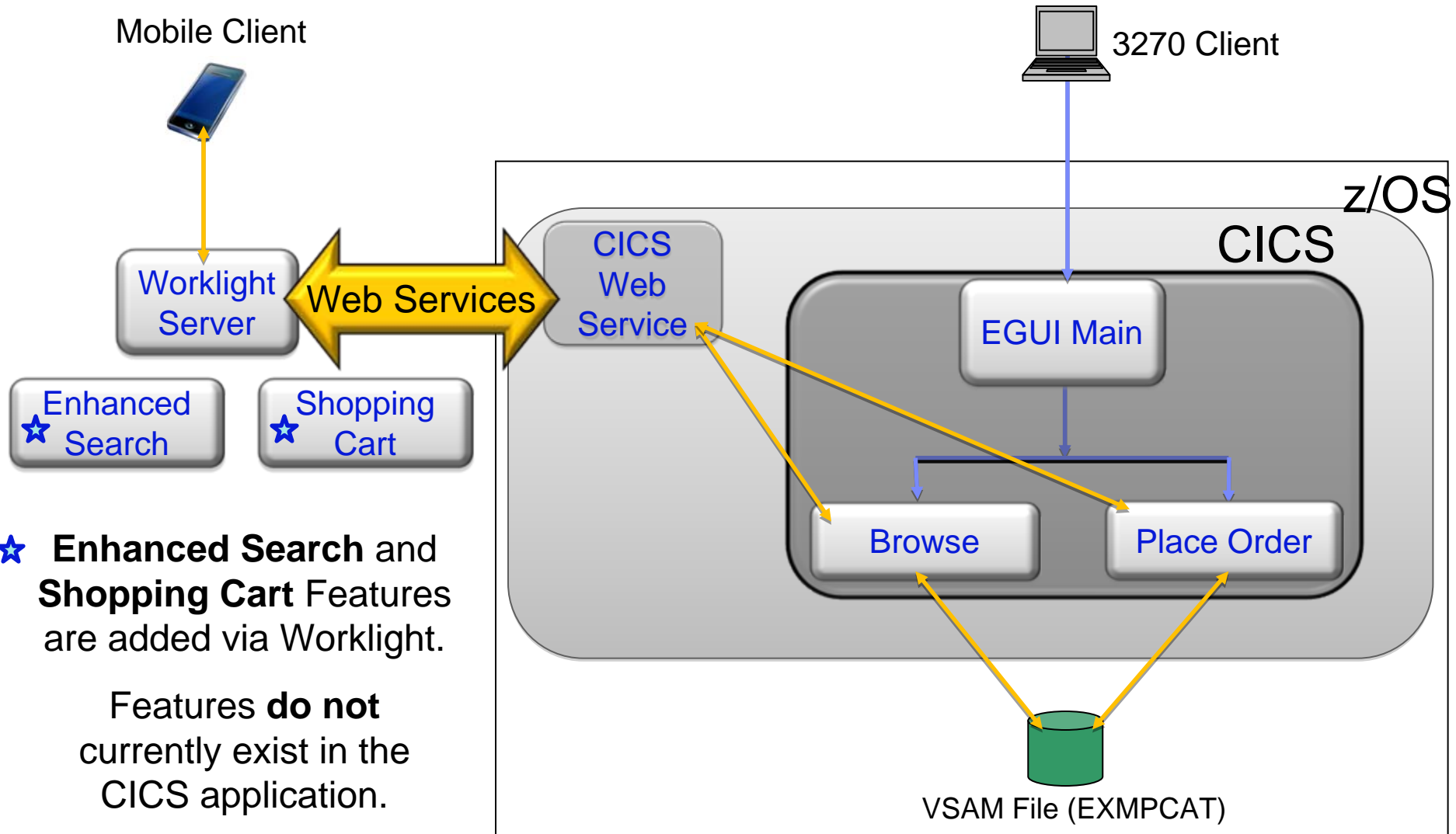
Item	Description	Cost	Stock	On Order
0050	Pencil with eraser 12pk	1.78	0067	000

Order Quantity: 3  
User Name: George  
Charge Dept: CICSUSRS

# CICS Connectivity Options



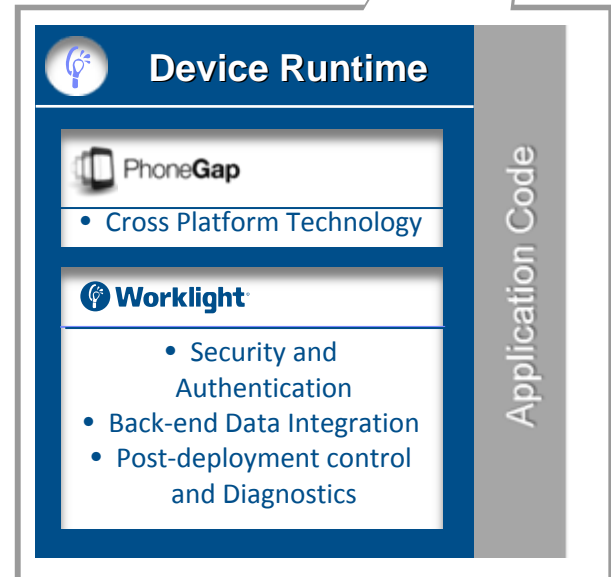
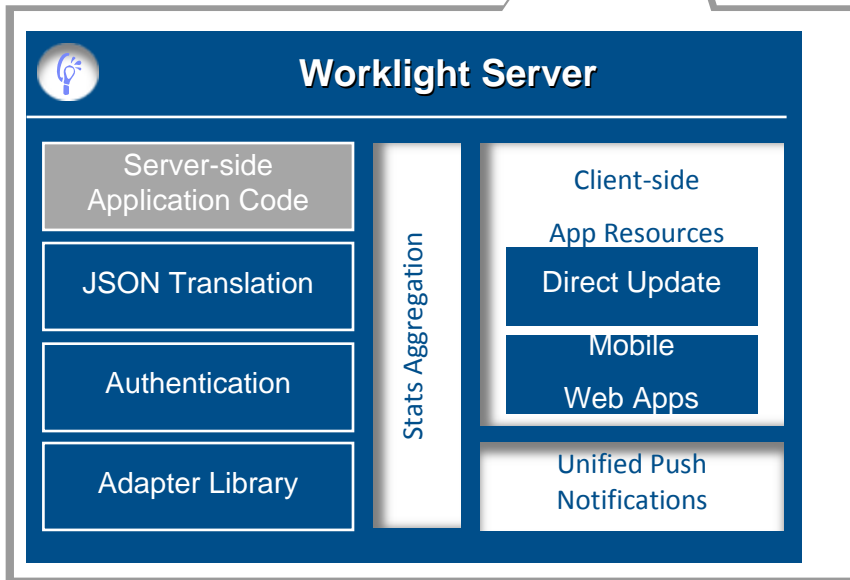
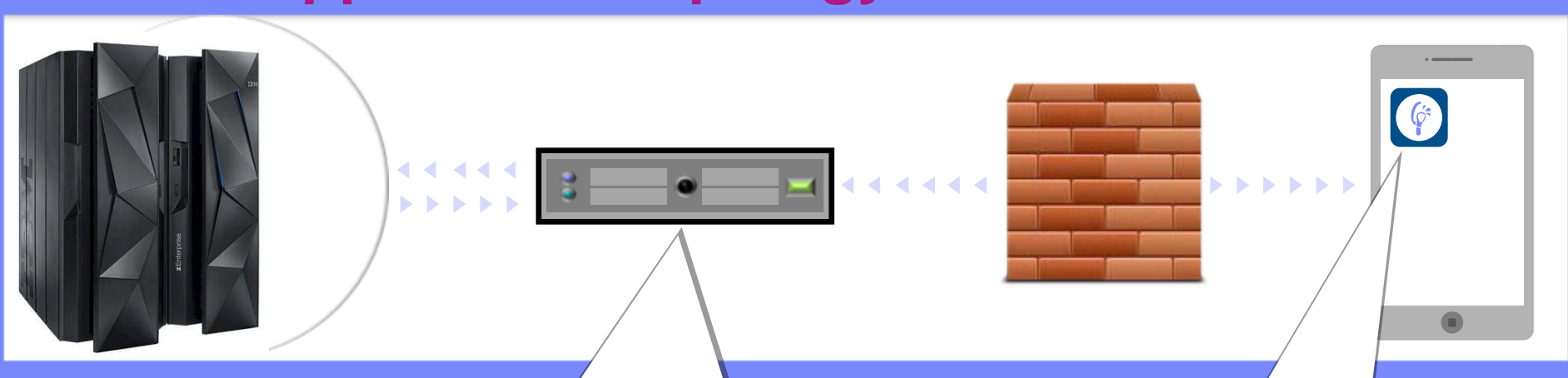
# CICS Web Service Enabled App



★ **Enhanced Search** and **Shopping Cart** Features are added via Worklight.

Features **do not** currently exist in the CICS application.

# Overall Application Topology





# MOBILE APP DEMO