On device development with Oracle’s Mobile Application Framework
Simplifying Mobility

Frédéric Desbiens
Principal Product Manager
Oracle Mobility and Development Tools
September 12, 2014
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Program Agenda

1. The Oracle Mobile Platform
2. MAF Overview
3. User Interface
4. Data Access
5. Security
The Oracle Mobile Platform
Oracle’s Mobile Strategy

Build Your Own Apps

Oracle’s Mobile Apps

MOBILE SECURITY

ORACLE MOBILE PLATFORM
The Oracle Mobile Platform

- Develop cross platform, multi channel and multi device applications
- Integrate data and services
- Secure information uniformly
- Deploy & Manage cloud and on-premises for multi-channel delivery
Overview
"A guiding principle for MAF is to eliminate the barrier of entry to mobile development for enterprise programmers, making them productive as early as possible."
Concept

Oracle Mobile Application Framework

Java
HTML5

Built On Standards
Use Existing Skills Set
Main Features

- Simple and Extensible UI
  - Components
  - HTML5/JavaScript
- Java for business logic
- Local SQLite database with Encryption
- Integrated Security
- Full Access to Native Device Features
A choice of tools
Oracle MAF Architecture
Oracle MAF Architecture

Mobile Device

MAF Device Native Container

Servers
Oracle MAF Architecture

Mobile Device

MAF Device Native Container

Java VM

Client Logic

Servers
Oracle MAF Architecture

MAF Device Native Container

HTML 5, CSS3 & JavaScript Rendering

Web View

AMX Feature
- Components
- Controller

Local HTML Feature

Remote URL Feature

Java VM
- Client Logic
- JDBC

SQLite
- Encrypted DB

Servers
Oracle MAF Architecture

MAF Device Native Container

HTML 5, CSS3 & JavaScript Rendering

Web View

AMX Feature

Components

Controller

Local HTML Feature

Remote URL Feature

Apache Cordova Plugins & APIs

Java VM

Client Logic

JDBC

SQLite

Encrypted DB

Device Services

Mobile Device

Servers
A few of our own MAF apps

PeopleSoft Campus Solution
Fusion TAP HCM
EBS Timecards
Pillar Axiom Anywhere
EPM Mobile
JD Edwards EnterpriseOne
User Interface
"MAF by design gives developers the choice to take a **productive declarative** route, or the **flexible programmatic** path. The user interface development is no different, the **choice is yours**."
User Interface Options

1. Local AMX Page
2. Local HTML Page
3. Remote URLs
Local AMX Pages

- JSF-like file built visually in JDeveloper
  - Component based framework
  - 80+ prebuilt “super” components
  - Rich data binding layer
- Generated into HTML/CSS/JS on device at runtime
  - Cross platform compatible
- Oracle is responsible for:
  - Device compatibility, support, upgrades
Local HTML Pages

• Any HTML5/CSS3/JS
• Total flexibility in design
• Integrates through JavaScript APIs with MAF
• All responsibility lies with the developer including compatibility, support, upgrades
Remote URL

• Call/embed remote websites

• Mix ‘n match local & remote content
"Application Mobile XML" (AMX) Page

- Define the UI of a single page
- XML format
- Design time support
- Owned by a MAF feature
- Can be bundled in task flows
AMX UI Component Based Framework

- 80+ prebuilt AMX components
- Mimic device look & feel
  - Oracle ensures cross platform support
- Can be bundled into reusable fragments
- No JavaScript skills required
- Support data binding
  - Can bind to data controls or Java beans
• Buttons
• Buttons and Links
- Buttons and Links
- Text and Selection
• Buttons and Links
• Text and Selection
• Data Views
- Buttons and Links
- Text and Selection
- Data Views
- Layout Components
• Buttons and Links
• Text and Selection
• Data Views
• Layout Components
• Operations & Behavior
• Buttons and Links
• Text and Selection
• Data Views
• Layout Components
• Operations & Behavior

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.
• LEDs & Gauges
• LEDs & Gauges
• Charts

![Image of bubble chart]

dvtm:bubbleChart
• LEDs & Gauges
• Charts
• Maps

dvtm:geographicMap
Data Access
"A challenge for mobile developers is integrating the numerous services & APIs to access data. MAF proposes a simplified approach through the introduction of data controls."
Data Controls

• Data abstraction layer
• Represent data sources
  – Local Java Beans
  – Remote SOAP/REST web services
• Expose
  – Collection
  – Attributes
  – Methods & operations

• Created via wizards and declarative editors
  – Metadata via DataControls.dcx
• Allow easy drag & drop of UI components
  – Data bindings automatically created
  – No/little UI programming intervention required
Data Control Options

MAF

Year
Region
Value

MAF UI Binding Layer

POJO Data Control

WS Data Control

POJO Data Control

REST Data Control

POJO Data Control (REST Adapter)

SQLite

SOAP

XML

REST

XML

JSON

Mobile

Remote

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.
MAF Binding Layer

- Declarative UI / Data Binding
- Available in MAF AMX features
  - Productivity layer in MAF
- Provides the actual data control "abstraction layer"
  - Exposes a consistent developer API

- The binding consists of XML at design time and Java objects at runtime
- Enables declarative and visual UI development
  - Drag and drop from data control panel
"MAF attempts to accommodate all types of web services and their different payloads, both declaratively and programmatically. It's a choice between productivity or flexibility."
MAF Web Service Support

1. SOAP with XML payloads
2. REST with XML payloads
3. REST with JSON payloads

- Best design time support via SOAP
- REST may be your preference, but
  - Mobile developers can't always dictate what service platform they consume
  - Lack of WSDL for REST discounts a declarative solution in some cases
  - Only remaining option is programmatic
The need for a local database

- High Availability
- Data Integrity
- Efficiency
Why SQLite?

- ACID
- SQL 92
- Self-contained
- Zero configuration
Main limitations

**Concurrency**
- Only a single connection can write
- Connections wait for the write lock

**Data types**
- Data integrity not checked

**Features**
- No stored procedures
- No special handling for XML/JSON

**Security**
- Authentication requires encryption
- No privileges (GRANT & REVOKE)
Security
Authentication
Supported modes

- HTTP Basic
- Federated SSO (Web SSO)
- Mobile and Social
- OAuth
Authentication
Types

HTTP BASIC
MOBILE AND
SOCIAL

Remote
Local
Hybrid
Authorization

<table>
<thead>
<tr>
<th>Authorization</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Roles</strong></td>
<td><strong>Privileges</strong></td>
</tr>
</tbody>
</table>

Component

Task Flow

Feature
Access Control Service

- A REST web service you implement yourself
- Must process POST requests
- Returns roles and privileges for the user as specified in the requests
- Not needed when using Mobile and Social authentication
Oracle’s Mobile Security Solution

Securely Separate And Manage Corporate Apps And Data On Devices

- Secure Container for App Security and Control
  - Separate, protect and wipe corporate applications and data
  - Strict policies to restrict users from viewing/moving data out of container
  - Consistent support across multiple mobile platforms

- Secure controls and management for enterprise apps
  - Secure communication with enterprise application servers
  - Corporate app store

- Extend IDM services to avoid redundancy and overlaps
  - Common users, roles, policies, access request, cert etc.
  - SSO for native and browser apps
  - Risk/policy based step up and strong authentication
Secure Container/Workspace for App Security and Control

- Separate, protect and wipe corporate applications and data on mobile devices
- Strict policies to restrict users from viewing/moving data out of container
- Consistent support across multiple mobile platforms
Secure Container/Workspace for App Security and Control

• Easily add new applications without requiring source code access or detailed coding
• Decouple security deployment & app development
  – Injection-based approach. No SDK.
  – Single sign-on and Data Leakage Prevention
Conclusion
Summary

- Oracle MAF is a **pillar** of the Oracle Mobile Platform
- Write once, deploy unmodified to iOS and Android
- **Productive** UI development
- Flexible data access through abstraction
- **Comprehensive** security
Upcoming features
What we aim to deliver in the next 12 months

• New AMX components
• Java 8 JVM
• Integrated data synchronization
• Tighter integration with Oracle ADF
• Support for Windows tablets and the Windows Phone OS
Let’s keep in touch!

Twitter
twitter.com/BlueberryCoder

Official blog
blogs.oracle.com/blueberry

Oracle Mobile Platform channel
www.youtube.com/user/OracleMobilePlatform
Hardware and Software
Engineered to Work Together