Air2Web Development Environment

Considerations and Capabilities White Paper

Air2Web
250 14th Street N.W.
Suite 4001
Atlanta, GA. 30318

Telephone 404-815-7707
Fax 404-815-7708
E-mail info@air2web.com

. **Table of Contents**

Overview…….

1. **Wireless Application Issues**
   - **1.1 Integration**
   - **1.2 Device Support**
   - **1.3 Interactivity**
   - **1.4 Infrastructure**

2. **Air2Web Development Architecture……**
   - **2.1 Application Databases**
   - **2.2 Provisioning**
   - **2.3 Device Catalog**
   - **2.4 Carrier Database**
   - **2.5 Message Rendering**
   - **2.6 Datafeed Processing**
   - **2.7 Input Processing**
   - **2.8 Event Processing**
   - **2.9 Message Delivery**
   - **2.10 Scheduling**

3. **Air2Web Development Environment……..**

   - **3.1 Open Standards Environment**
   - **3.2 High-level APIs**
   - **3.3 Simulation and Testing**
   - **3.4 Traffic Monitoring**
   - **3.5 Security**
   - **3.6 Billing and Reporting**
   - **3.7 Internationalization**

4. **Summary of Benefits..……..**

Overview

Developing an interactive, wireless application from scratch can be a complex and difficult undertaking--even for the most experienced development team. Creating a solution that is scalable, reliable, robust and flexible enough for a broad range of users requires understanding and accommodating multiple carrier service types, a wide assortment of end user devices and formats, various security standards and mechanisms, and a range of data types.

Air2Web has packaged a complete set of mobile and wireless services to assist developers and organizations in quickly and efficiently designing, creating and implementing wireless applications. Whether these applications are complex and interactive or as simple as text broadcasts, Air2Web simplifies the development and implementation process by providing developers a complete selection of tools and hosted services.

Developers can utilize libraries of application templates, pre-packaged functions and services or write to high-level APIs to streamline development. Air2Web hosts applications and services through state-of-the-art data centers that provide a scalable and reliable application infrastructure. In addition, Air2Web is already connected to all major regional, national and international carriers.

The Air2Web platform accommodates a wide assortment of currently available wireless clients including 1- and 2-way pagers, 1- and 2-way SMS phones, WAP (or Web) phones and PDAs. Developers design and develop a single application and the interaction with multiple device types is handled automatically.

In a fast changing market where standards are evolving, new devices are rapidly coming online and the demand for multiple types of mobile information is exploding, Air2Web jumpstarts development and provides hosted services for continued operation. The benefits to organizations requiring mobile applications include:

- rapid implementation
- lower development costs
- simplified management
- ability to keep current with latest technology
- eliminating self-managed IT infrastructure

Wireless Applications Issues

The high growth and wide availability of wireless devices such as pagers, cell phones and personal digital assistants (PDAs) has created a communications infrastructure that can be leveraged to provide unique and powerful interactive mobile applications. The uses for this capability are virtually unlimited but include some of the following examples: mobile banking, financial market transactions, delivery tracking, remote auction participation, wide area group scheduling, sales force automation, purchase authorization, sports score and weather tracking, etc., plus hundreds of business specific applications. According to Gartner Group, mobile phones will be the most common device accessing the Internet worldwide by 2005.

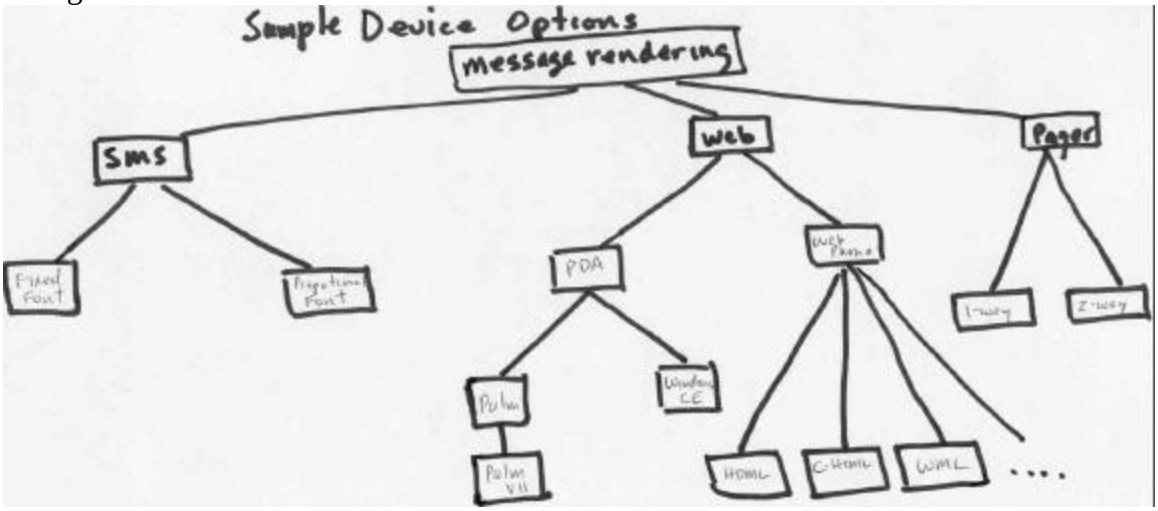
There are several significant issues that must be addressed when creating interactive mobile applications. These issues include integration between Internet applications and wireless carriers, dissimilar device support, interactivity, development tools and infrastructure issues such as scalability and security.

Integration of Web/IT Infrastructure to Wireless - Mobile web applications by nature involve integrating two different networks: the Internet and one or more wireless carriers. Since these two networks have evolved from separate origins with different data types, dissimilar protocols and separate standards, the integration of technologies can be complex. Data from one protocol must be converted to the other; graphics and text must be treated differently depending on the network and the device; end-to-end security must be established; and a management system must be implemented to monitor status and usage.

Integration of these two networks is generally handled through a web-to-wireless gateway. The primary challenge with a 'self-developed' gateway is its static nature and rapid obsolescence. As new standards or devices become available, the gateway must be continuously modified and updated to stay current. The arrival of a new wireless carrier service or a new device can dictate the complete redevelopment of a self-developed gateway. A service organization that establishes multiple carrier connections, maintains a common gateway and consistently monitors and enhances the connection has clear advantages over self-managed systems.

Support of Diverse Wireless Devices - A major challenge in developing any type of widely used application is accommodating disparate end-user devices. The four general classes of wireless devices (pagers, digital cell phones, WAP phones and PDAs) all have varying levels of functionality when it comes to receiving data, interactive responses and data presentation. For an application to be functional across various clients, a solution must be implemented that recognizes a particular device's capabilities and presents the application output in context of that particular device. For example, an application that broadcasts stock prices would send only periodic price figures to a pager but may send a graphic stock chart to a wireless PDA. Some devices allow 2-way interactivity while others only support 'pushing' information (i.e. Web phone vs. 1-way pager).

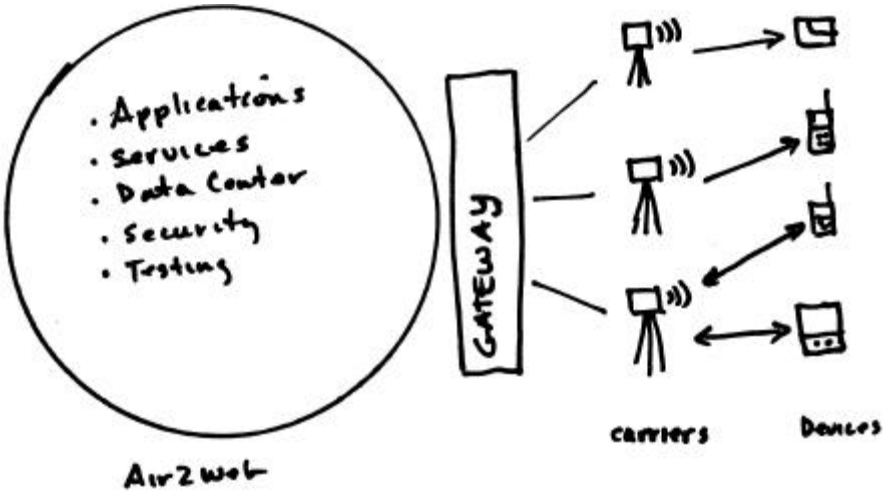
It is not feasible to develop individually separate applications for each device. Air2Web lets you develop applications once which can be used from/for any device. The application does not have to change as new devices emerge. Air2Web adds support as new devices come along making it possible for developers to 'create an application once - and have it available on a comprehensive collection of devices'. Spreading the development and maintenance costs of individual device support over many applications provides an economy of scale that justifies continuous improvement as new technologies emerge.



Interactivity - Mobile applications can differ widely in the level of interactivity required. For basic applications, a simple text transmission to the remote device is sufficient. Other applications however, require that there be an interactive session between the application and the client with menu selections, text or numeric data and even pass codes being keyed in at the client for transmission back to the server. More sophisticated applications must accommodate the ability to provide intelligent dialog between the end user and the server application that may include entering numbers or text, selecting from menus or even capturing voice commands.

One of the challenges with interactivity is that different remote clients support different levels of interactivity. Another challenge is the degree or level of logic that a device is designed to support. A Web phone for example, can actually display stacked menus with navigable screens for selection. Some devices can parse text messages and perform operations (such as call a phone number) while devices such as PDAs can actually process received input and then return computed results to the application. Understanding all the classes of interactivity and the information presentation capabilities for each device requires a heavy investment in developer research and training.

In addition, Air2Web not only enhances two-way communication for interactive devices but also provides base level interactivity on some devices that are inherently one-way. For example, an application may generate output in the form or a phone number and send it to an alpha numeric pager; calling this phone number from any phone will initiate an interactive response. Enabling the large installed base of one-way devices to 'talk back', allows organizations to take advantage of the worldwide base of existing users.



ronment

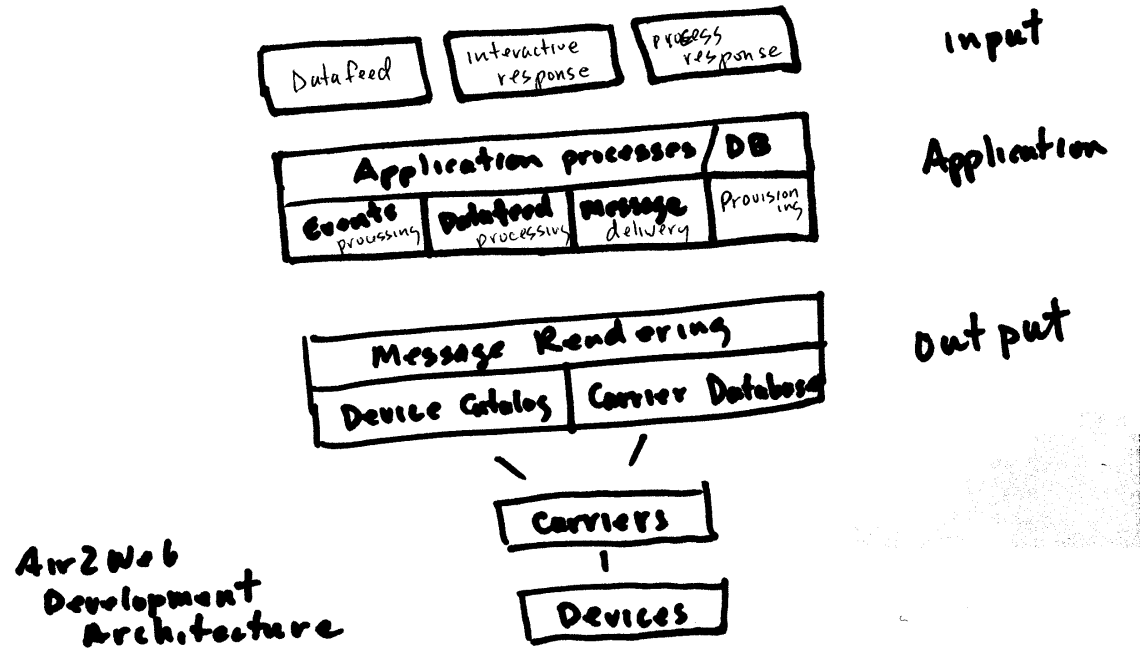
Infrastructure - To create a self-managed Internet/wireless application would require acquisition of hardware and software, setup and integration, Internet connectivity, carrier connectivity, plus the onsite staff for management and administration. This effort, in addition to cost, often requires at minimum several months of implementation, integration and testing before it becomes operational.

In addition, the ongoing effort for maintenance, support, hardware and software upgrades and the cost for utilities and physical space can be significant. Air2Web provides a complete IT infrastructure for Web and wireless applications. For most applications, there is never a need for onsite hardware, software or IT infrastructure. Air2Web's data center is world class with state-of-the-art hardware, software, security and Internet connections. Redundant systems and triple power supplies are in place; backup solutions are maintained and facilities are secure from electrical, water and seismic threats. All facilities are also securely guarded with strictly enforced authorized-only entry.

Air2Web Development Architecture

Air2Web is emerging as a premier provider of hosted solutions and services for the development of wireless applications. Using resources available from Air2Web, developers can quickly create, test and deploy interactive mobile applications that work with any wireless device. Air2Web provides a hosted infrastructure for applications as well as a collection of utilities and services that can enhance and streamline any specialty or mass market mobile solution.

The Air2Web platform is designed to provide all of the tools, utilities and services that are required to complete a wireless application development project with a minimum amount of resources in the least amount of time. Underlying this capability is the Air2Web platform architecture that consists of the following subsystems: application database, provisioning, event processing, device catalog, carrier database, input processing, datafeed processing, message rendering, and scheduling.



Application Database - The application database consists of the logic, processes and data storage that are part of any custom application. The creation of an application includes the ability to receive or generate information that is 'published' to subscribers with wireless devices.

Since custom applications vary significantly, Air2Web has developed an abstracted level of application functionality for use by developers. The basic components of an application include the following:

- template interface - custom or pre-packaged application interface with fonts, colors, etc.
- member definition - the categorizing of subscription information according to members or groups (i.e. - a baseball application member would be a specific team and a group would be the 'National' or 'American' league)
- topic definition - a topic definition specifies 'what' information is to be transmitted (i.e. - sports score topics might be 'send scores at end of each inning' or 'send only final score')
- event definition - events are triggers that initiate runtime processing in the system
- publication definition - the publication definition defines the format(s) of data that is published to various wireless devices

Using these high-level components in conjunction with other Air2Web subsystems, application developers can create any type of 'push' or 'interactive' mobile application.

Provisioning - Air2Web's provisioning subsystem allows publishers (customers of Air2Web's services) to quickly create the interface that will allow subscribers to sign up for an application's services. Subscribers enter profile information, select their wireless

device, select member and topic notifications and configure on-demand functionality. The end result of a subscriber going through the provisioning process is a 'subscription'.

Publishers can create the provisioning interface in two ways. The first is to write a custom interface using Air2Web's high-level API; the second is to use predefined, customizable interfaces or 'templates' that can be deployed extremely fast.

Device Catalog - A major barrier in developing wireless applications that work across multiple devices is customizing input/output for each device type. Air2Web has implemented a comprehensive device catalog that coordinates information publishing formats with device types and models. This catalog includes the data definitions for each of the following device types:

- 1-way SMS cellular phone
- 2-way SMS cellular phone
- 1-way pager
- 2-way pager
- Personal Digital Assistant (PDA)
- Web enabled phone (HTML, WML, C-HTML)

Carrier Database - In addition to accommodating multiple device types, wireless applications must work across multiple carriers. Air2Web has established relationships and maintains connections with over 460 different regional, national and international (North America, Europe, Asia, Latin America and Australia) phone and wireless carriers. The interfaces, connections and ongoing maintenance with each of these carriers is completely handled by Air2Web. Different network standards are supported such as TDMA, CDMA, iDEN, and GSM. Developers are insulated from the intricacies of each of these separate services and only need to program to Air2Web's open interface.

Message Rendering - The Air2Web messaging rendering subsystem allows a publisher to optimally present data on each supported wireless device. The subsystem supports a 'meta' markup language that allows publishers to minimize the number of template definitions necessary to present data appropriately on various devices. For example, the publisher creates a template that works for all SMS devices that have proportional fonts and another template that works for all SMS devices with fixed fonts. The message rendering subsystem automatically takes the high-level XML data and renders it to the appropriate device specific format. Message rendering also includes the growing area of voice support. Many devices are now able to accommodate the receipt of voice files or text-to-speed converted output.

Datafeed Processing - Many mobile applications consist of capturing datafeed information that is generated from external sources and forwarding all or part of it to mobile devices (i.e. stock prices). The datafeed subsystem is the external interface for data content updates received by the application. External data is translated to an Air2Web format that can then be forwarded as device specific representations.

Input Processing - Air2Web applications are interactive across all devices. This interactivity implies that the system is capable of receiving input from subscribers using dissimilar devices. The input processing subsystem simplifies development by resolving input from various devices to a common format that is consumed by the application. This insulates publishers from the differences between device platforms making it possible to define the information that needs to be captured from the subscriber once, and apply that definition across all supported platforms.

For example, an application that calculates the shipping cost between two postal codes requires input of the two codes. Using a 1-way SMS device, the two codes would be captured using DTMF detection (dialing numbers on a phone); using a Web phone, the codes are entered via edit boxes in the phone's browser. Regardless of the input method, the input processing subsystem collects the information and hands it off for processing. Input data types include:

- Postal codes
- Phone numbers
- Yes/No Boolean values
- Personal identification numbers (PINs)
- Integers
- Decimals
- Enumerated types (lists of choices)

Input also can be in the form of voice commands. The ability to accommodate both alpha-numeric as well as human voice input with speech-to-text conversion is handled by the input processing subsystem.

Event Processing - Event processing triggers the output of all published information. Events are received from various input sources including data feeds, internal processes and wireless devices. The Air2Web event processing engine is capable of sustaining high transaction loads and supports the following event types:

- publication update
- publication purge
- input processing
- scheduling

Message Delivery - The message delivery subsystem provides a stable, fail-safe system that handles a high volume of messages sent to end clients. This includes the ability to accommodate the following features:

- notification of message delivery failure
- message logs
- sending undeliverable messages to alternate address
- monitor performance stastics
- support for gateways (SMTP, GSM, SMPP, HTTP, etc.)
- message canceling and aging

Scheduling - The scheduling subsystem is an event engine that initiates operations based on time, events or conditions. The scheduling subsystem initializes new processes, initializes existing persisted processes and deletes old processes. Once a scheduled event is triggered, the process scheduler may initiate the event via a message into the message queue or interact with other system services.

Scheduled processes have the following parameter variables: start timestamp, stop timestamp, and process interval. Start and stop timestamp are creation and termination variables and process interval is the duration of time between the execution of the scheduled event (i.e. once, daily, weekly, etc.).

Any or all of the components mentioned above can be used as part of a mobile application. The underlying functionality provided by the Air2Web collection of subsystems makes it easy for developers to design and 'assemble' applications rather than build from scratch.

Air2Web Development Environment

In addition to an efficient development architecture, Air2Web has created a development environment that again simplifies and expedites the creation of custom wireless applications. Using open standards technology, high-level APIs, simulation and testing tools, and traffic monitoring plus providing security, billing and reporting tools and built in internationalization capabilities all contribute to efficiency in development, testing and deployment.

Open Standards Environment - Air2Web solutions are built to open standards for both Web and wireless carrier networks. In the Internet arena, this includes HTML, HTTP/S, SSL, XML, CGI, PERL, Java Beans, Java Server Pages and industry accepted security standards. The advantages of using open standards include platform independence, location independence and ease of integration with other applications and solutions. Using open standards also helps extend the life of a solution and makes it easier to transfer testing, maintenance and enhancements to other developers familiar with open standards-based code.

High-level APIs - For developers using Air2Web solutions, most labor-intensive development is already done. Air2Web provides high-level abstractions of low-level functionality that a developer can simply call and provide parameters. Using pre-packaged functionality and modular applications, developers can quickly assemble applications without having to generate code from scratch.

The use of n-tier architecture components (such as Java Beans) makes it possible for publishers to focus on the business processes rather than the application development details. Resources are focused on making sure the information flow and results are achieved while applications are created using modular, reusable, pre-packaged applets and components.

Simulation and Testing - The successful deployment of any application depends on adequate simulation and load testing. Air2Web provides testing services that allow developers to simulate interaction, device output, load stressing, and carrier connections. Using these tools, any application can be rigorously tested before deployment to ensure that initial 'live customer' experiences are positive and free from business damaging effects.

Traffic Monitoring - Successful Internet applications generate high traffic that can in some cases be used as a source of revenue. Air2Web provides a set of services that simplify the monitoring and management of applications and provide statistical summaries that can be used to verify ad revenue potential or detail traffic and click thru. Partnerships are in place to enable sponsorship and track ads (Media Metrix, Nielsen/NetRatings, AdForce, Hitbox).

Security - The process of securing wireless application information requires the protection of data at multiple segments. Air2Web has developed a sound security strategy for protecting data through the application segment, to the Air2Web gateway, to the carrier gateway, on to the remote device and back again. Security is based on the best Internet and wireless carrier technology and includes authentication, encryption, data masking, security protocols, PKI, and a fully secured data center. (see Air2Web white paper on 'Wireless Application Security')

Billing and Reporting - Application Service Providers that create solutions built on Air2Web's services often require the ability to track and bill based on usage. Air2Web provides a set of services that allow providers to track and bill based on message volume or character quantity, air time, call volume, call location, page hits, session length, etc.

Internationalization - The growth of application enabled mobile devices is actually increasing faster internationally than in the U.S. As a result, applications need to be enabled for multiple languages in order to take advantage of this growth. All Air2Web applications and services are double-byte enabled which allows easy customization for any foreign language through the use of standard translation tables.

Summary of Benefits

Organizations seeking to implement mobile wireless applications can benefit significantly by using Air2Web's existing development services and established infrastructure. The advantages of using a hosted service approach from an established service provider can be categorized under quick deployment, simplified management and reduced costs.

Quick Deployment - By taking advantage of pre-developed and pre-tested components and an already established infrastructure, application developers can cut development, testing and deployment time significantly. Applications can be assembled rather than written from scratch. Existing simulation and testing suites assist in quickly eliminating bugs. The availability of an existing IT infrastructure eliminates the time required to procure, setup, install and connect hardware and software.

Simplified Management - Air2Web handles all of the intricacies of complex device and carrier connection management. Management of connections, devices, security, gateways, Web traffic and applications can be transparent to a specific application developer or owner. In cases where an application requires specialized management, additional high-level management interfaces can be surfaced for customized use.

Management is also simplified by the fact that Air2Web handles all of the changing variables associated with multiple device types and multiple carriers.

Reduced Costs - In addition to cost savings from quick time to market and simplified management, users of Air2Web services benefit from the economies of scale of a shared infrastructure. The actual cost of a world class data center with fail over capabilities, multiple connection points and state-of-the-art hardware is divided among multiple parties. Customers gain the advantage of top level service and infrastructure but only pay for actual resources in use.

In summary, creating any mobile, wireless, interactive application whether it is horizontal and mass market or vertically specialized and unique will be faster, simpler and less costly when using Air2Web’s development platform and hosted services.