J2ME Case Study

David Moskowitz
Productivity Solutions, Inc.
Agenda

- The Vision / Mission
- Building thin clients
- J2ME
- Security
- The economic justification
- The extraordinary results

This is a companion presentation to "Patterns Are for More than Code: Building a Real-life Web Service-based Application". The economic results in this presentation apply to both.
Vision

- Web Services can be extended...
  - to provide information-focused tools and...
  - simultaneously expand the reach of Business Intelligence (BI).

- The goal for the implementation: Make information available everywhere, anytime.
  - Mobile enablement of the workforce is not just a business requirement it is also (and more importantly) a business differentiator.
  - Traditional access models no longer sufficient.
Mission

- Start the initial pilot with field sales people
  - Furthest away from the source
- Sales automation including BI on wireless PDAs
  - (phase 2 added sales support personnel and others).
- The application will include Java enabled phones (and other devices), as the installed base of suitable devices grows.
Initial Decisions

- Information everywhere anytime means mobile devices.
- Two choices
  - Daily sync with data on the mobile device
  - Wireless request with current data
- The initial goal was to craft a Customer Relationship Management (CRM) application to help build share of customer.
Wireless Value Proposition

- Wireless adds value in more than person interconnected
  - Manage supply and demand
  - Interconnections that add value
  - Timely response to the customer
  - Adds a totally new dimension to high-tech, high-touch, customer relationship management

- You win in the wireless space by making information available where it needs to be.
# Mobile App Pieces

<table>
<thead>
<tr>
<th>Application</th>
<th>CRM, ERP, Inventory, Vertical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Devices</td>
<td>PDA, Laptop/tablet, phone, RIM, Symbian</td>
</tr>
<tr>
<td>Connectivity</td>
<td>Wireless, Wire line, VPN, RAS, Dial-up</td>
</tr>
<tr>
<td>Mobile Middleware</td>
<td>Data Sync, e-mail sync, SysMgmt, Personalized</td>
</tr>
<tr>
<td>Integration points</td>
<td>Database servers, File/Web servers, Exchange/Domino</td>
</tr>
</tbody>
</table>
Wireless Pros and Cons

- Increased sales
- Decreased costs
- Improved customer service
- Competitive advantage
- Rapid ROI

- Coverage
- Reliability
- Standards
- Speed
- Costs
- Small devices add constraints
- Unpredictable connectivity
The Initial Questions

- What if the requested service isn't available?
- How are messages correlated?
- What if there isn't a connection available?
- What if the response is delayed?
- What if...?

- See the companion presentation for more information about these areas.
How Do We Provide…?

- Mobile users with the ability to seamlessly roam from device to device...
- Access to the same applications
  - With some commonality of information.
- Synchronized data to a mobile device for access at a later time or to operate even when (occasionally?) disconnected.

Can we meet these requirements, today?
Driving Ideas

- It's about integrating disparate databases to gain real Business Intelligence.
- The information is there, the problem is integration analysis, access, and presentation
  - The only parts we can do on the mobile device are transaction preparation and presentation – and that is very limited
- Need a way to measure ROI & determine payback
Driving Ideas

"With the correct use of information technology, we can create virtual databases that will enable XYZ to become much more competitive because we won't have to constantly negotiate with constituencies."

- Change customer experience – CRM, billing, delivery, demonstrations, choice and fulfillment
Driving Services

- Notification: alert, change or request
- Location-based: GPS
- User-context: If it's Tuesday: on-site
- Device services: device specific transforms and/or device specific content.
- Synchronization: content sent to and from the device (typically app messages or e-mail)
- These services interact with each other
Tools Used (Partial List)

- J2ME
- WMA (http://java.sun.com/products/wma/)
- J2EE (on the server)
- Various emulators (some 3rd party, some home grown).
- TLS (RFC 3546) and SSL
  - Also http://www.bouncycastle.org/
- Palm OS, Motorola, Nokia, RIM
- Wi-Fi, Verizon Wireless and others
Building Thin Clients

- Don't even think about doing a lot
  - When the developers first started, the "little list" of neat things to do was huge...
  - Then reality...
- Web programming model revolves around application servers
  - Clients render HTML
  - Too many constraints for a small device
Building Thin Clients

- What about WAP and WML?
  - Tied too closely to current Web app model
  - Limits the messages

- Requires specialized tools and languages
  - J2ME and emulators (the real device is too slow and cumbersome for extensive testing)
  - Minimal functionality means less complexity and faster time to market
  - Functionality specifically tailored to need
Building Thin Clients

- Lower bug count – not without challenges to create
  - Most problems exist on the server
  - Simpler programming model
- Lower admin costs – easier to deploy
- Better application and data security
  - Most data still on the server
- Idiosyncrasies of wireless connectivity & limitations of current technology devices make building wireless thin clients a hard problem.
J2ME

- Evolving faster than desktop/server Java
- Suitable for small devices with limited capabilities (cell phones and PDAs).
- J2ME devices have:
  - Limited input capabilities (no keyboard!)
  - Small display sizes
  - Restricted storage memory and heap sizes
  - Slow CPUs
J2ME Configurations

- **Connected, Limited Device Configuration (CLDC)**
  - Specifies Java environment for mobile phone, pager, and PDA class devices
  - Usually wireless

- **Connected Device Configuration (CDC)**
  - Specifies Java environment for digital television set top boxes, high end wireless devices, and automotive systems.
  - CDC devices may be wired (DTV cable, *etc.*)
J2ME Profiles

- Supplements a configuration to provide capabilities for a specific vertical market or device type
- Mobile Information Device Profile (MIDP)
  - Application runtime environment for devices based upon CLDC
- Foundation profile
  - Non-GUI networked devices based upon CDC
- Personal basis, personal & RMI profiles
  - Basic graphics – next generation Personal Java
CLDC

- Target devices:
  - 160KB to 512KB total available memory
  - Limited power (usually battery)
- Limited (intermittent) connectivity to a network (usually wireless)
- Constrained UI (very small screen)
- CLDC reference implementation built using the KVM
MIDP

- Targets mobile two-way devices implementing J2ME CLDC
- Profile addresses
  - Display toolkit, user-input methods
  - Persistent data storage using simple record oriented database model
  - HTTP-based networking using CLDC Generic Connection framework
Implementation Overview

- The application is implemented as n-tier:
  - The user interface on the mobile device (client tier) provided by a MIDP application (or MIDlet).
  - The MIDlet talks to (not directly) a servlet in the Web tier to access the business logic tier.
    - After authentication – both one-way and two-way
  - The servlet also supports seamless access to the database tier through the JDBC API.
    - Servlet is actually a set of servlets as "Controller"
MIDP Applications

- Lifecycle: Paused, Active, Destroyed
- Packaging
  - MIDlets JAR files including
    - Class files of the MIDlet(s)
    - Resource files
    - Manifest with application properties
  - Application Descriptors (JAD files) accompany MIDlet JARs and provide deployment information (name, version, size, etc.)
MIDP UI Design Principles

- Mobile Information Devices are consumer products, *not* desktop computers
  - Constantly think of end user
  - Solve the *real* problem for the "real client"
- Use simple traversing and selection metaphor
- Must be usable in all devices
  - One handed, two handed, stylus operation
  - Screen (10s (100s) of pixels by 10s (100s) of pixels)
  - Some don't have a pointing device
- Look & behave consistently on any given device
Other Application Information

- **Security**
  - Both SSL and TLS used to encrypt messages
  - Depending upon types of requests (determined by state information) both 1-way and 2-way authentication used.

- **Transaction IDs built into messages**

- **Used J2ME Wireless Toolkit**
  - Includes Wireless Message API (WMA) for SMS
  - James Osbourn: "Writing Java Applications for Mobile Information Devices"
On- and Off-line Connectivity

- Service-based require "server-level" support
  - What if connectivity isn't available?
  - What if it's just the end point that's down?
- How much local data is (or will be) available?
  - Can local data be used?
  - How much data is pre-loaded (and how)?
- How much can be done on the mobile device while waiting for connection restoration?
  - Is the time of the user transaction significant (versus server-level timestamp)
On- and Off-line Connectivity

- Design app transactions to use message-based UI interactions.
  - Takes extra effort to manage optimistic locking & the transaction results based upon old data
    - Assumes that any commit has a chance to fail
    - Optimistic locking avoids deadlocks – critical in this type of environment
  - Submit both old and new data and let the service resolve conflicts (if any)
On- and Off-line Connectivity

- Local cache for read-only data?
  - Include time-out information if relevant

- Implement an off-line business service that has the same interface as the server
  - Use store and forward

- Add functionality to the UI to inspect outbox
  - In some cases, allow requests to be deleted
  - Raises security questions

- Let user persist state to memory card
**Economic Justification**

- Company examined various costs
  - Cost of sales
  - Cost to support field personnel
- Sales cycle time
- Stable (slight decline) market share relative to competitors because of response time
- New product introductions required extensive field training
- Bottom line: cut costs, increase revenue and productivity
What Does It Take to Do It?

- Commitment – the process can be arduous
  - Have to focus on the end results
- Need buy-in from every internal constituency
- Costs can be scary (in this case...)
- What is your guess?
  - Does it help you estimate if you know the new infrastructure cost was close to $900K for the h/w and support staff?
What Does It Take to Do It?

- Partner with multiple vendors
  - Including consultants
- Recognize the impact will be broader than IT
- Consider both TCO and TBO
  - Total Benefit of Ownership (what you're getting for what you're paying)
- Ask the right questions about your business
  - What business are you in?
    - Do your customers agree?
  - What can you do to anticipate customer expectations – in real time?
    - How do you know the questions they'll ask...?
Results

- 15% of sales electronic
  - Old system cost almost $60 more per order
- Lost or late orders cost $150 per to track
  - 2.57 missing or delayed items per 1000 items vs.
  - 3.98 items per 1000 items prior
- Reduced maintenance costs saves 11%
  - Almost $9M
- Cut costs & increase revenue exploiting data warehouse — about $10M/yr through better data analysis
Results

- Mobile office:
  - Wireless e-mail, IM, calendaring, order entry, support
  - Yearly cost savings per sales person: $24K
  - Facilities, separate communications, storage, support

- Wireless sales force automation
  - Increased sales 10 to 20 percent by allowing almost two additional orders per day per sales person

- Wireless field service automation
  - 20% more inspections/day; 50% less waiting for clients

- New customer experience and satisfaction
  - 12% more orders; each order 4% higher
Results

By finding canonical ways to express, capture and record transaction information, estimate savings of $40M

- Canonical expression reduces training costs by over 25%

- Bottom line: projected payback period was 2+ years
  - In actuality – less than 6 months
Results

- Not only did they save money, but, anytime, any place, anywhere to anybody actually helped them identify new business
- Turn information into an asset that produced $120 million between 7/2003 and 10/25/2003
  - Projected to grow, over the next 3 years to be over $1 billion.
Results

- This project involved change
  - Business process changes within the company to accommodate the new mission (information everywhere)
  - Development process changes to build software for mobile devices.
- Change is painful, sooooo...
  - ... don't make changes – experiment with new possibilities
  - Expect acceptance vs. participation
Questions & Thank you

- For more information

David Moskowitz
Productivity Solutions, Inc.
147 Ashland Avenue
Bala Cynwyd, PA  19004
Voice:  610-664-7726
Fax:  610-667-1798
E-mail:  davidm2@usa.net