Softwired

Technical Briefing

Silvano Maffeis, July 2001.

Improving the User Experience

Problem

Wireless platforms based upon markup languages such as WML have not proven very successful yet. For example, WAP users often complain about a poor user experience due mostly to poor content, long delays in accessing WAP pages, errors occurring when a device loses network coverage for a limited period of time, and so forth. In our opinion, just deploying WAP over faster bearers, such as GPRS, will not solve the problem. We believe the problem is more fundamental and relies in the fact that the "request/response" interaction model, which is used successfully on the wireline Web, was simply transferred to mobile devices, without taking into consideration that mobile devices are used completely differently than PC workstations at the office.

Solution

We believe that in the mobile world, user experience can greatly be improved by providing an interaction model in which the user does not have to go and look for content using a browser, but where the user is *alerted* when something interesting happens. Such an alert should take the user's preferences and geographic location into consideration, ensuring timely delivery of what an individual user considers to be critical or useful information WITHOUT the need to actively search for it.

User experience is further improved by installing customized applications on mobile devices, which allow the user to continue working with the mobile application even when disconnected from the

more

network. Those applications will be built using a "store-and-forward" messaging paradigm, which is provided through appropriate messaging middleware technology on the device.

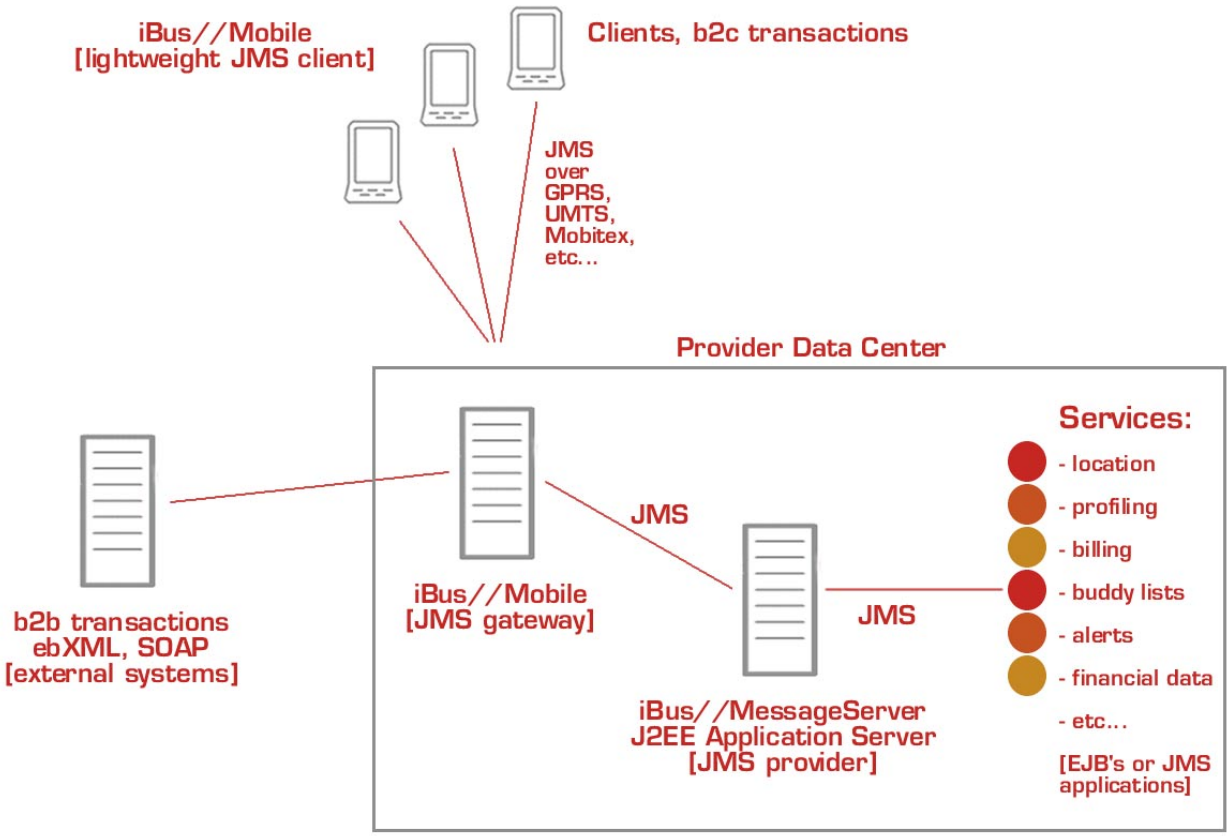
Examples of applications that take advantage of this messaging paradigm are workforce applications, allowing a mobile worker to keep on entering data into his mobile device, even when the device drops in and out of network coverage. The messaging middleware beneath the mobile application takes care of store-and-forwarding this information to a back-end server, as soon as network coverage is re-established. Other examples are mobile commerce applications, news tickers, messengers, and games.

A Standards Based Mobile Applications Platform

Softwired's approach to developing the aforementioned type of mobile application is to use Java technology straight through from the mobile devices to the application servers. The approach consists of deploying customized Java applications on the mobile device, and on the server computers hosting business information and "back-end" services such as location, customer profile, email and schedule integration and also billing services, for example. Then, we use a packet-oriented wireless bearer such as SMS, GPRS, or Mobitex to connect the mobile devices to the back-end servers in a flexible and scalable manner.

To make the interactions between the clients and the servers reliable and secure, we deploy standards-based Java messaging middleware technology on both the mobile devices and the servers. For this we rely on our iBus product family, industry's only end-to-end JMS (Java Message Service) middleware solution.

Other standard-components we integrate into our iBus platform are J2EE application servers such as BEA/Weblogic. The application server runs at the "back-end" and provides business logic and wireless services in the form of Enterprise JavaBeans (EJBs).



In summary, our approach consists of installing J2EE application server technology in the data center, J2ME applications on the mobile device, and in using JMS as a robust and scalable "communications backbone" between the devices and the servers. For more information about this type of mobile platform solution, visit:

http://www.softwired-inc.com/

Silvano Maffeis is CTO at Softwired AG. Softwired is a privately held, venture capital funded middleware software company headquartered in Zurich, Switzerland. Silvano Maffeis can be reached at maffeis@softwired-inc.com.