User Preferences of Web (Services) for Distributed Web Applications

Kangchan Lee, Seungyun Lee/ETRI

Web Services is currently emerging as a dominant technology for supporting e-Business automation and integration, it is also increasingly considered as a promising platform for inter-connecting devices in mobile and ubiquitous computing environment. And, by embedding Web Services into virtually any computing devices, it becomes possible for a device to discover and interoperate with other devices and remote services, establishing pervasive network of computers of all form factors and wireless devices.

When Web Services client is enabled in mobile device and engages in interactions with a service provider, it becomes necessary to define an interaction logic required for them. For this purpose, one can use a choreography language to specify the rules of engagement between the mobile client and the Web Services provider. In mobile service environment, however, connection may be lost and the mobile device may move into out-of-service area any time during a conversation, and the user may not be attentive all the time. These may prevent the conversation from successful completion particularly when the conversation is long-running or involves several user interactions. Accordingly, performing mere step by step execution of a choreography specification defined for the mobile client may yield unsatisfactory results.

So, it is need to specify the client-side preferences on how a client should interact with a Web Services provider. For example, in order to start the service, the device first needs to invoke the service by sending context data such as the authentication information, screen size, and media handling capability of the device to the service provider. For the sake of simplicity, we assume that only the screen size information is required. When the service provider receives the context data, it immediately checks if the configuration of the device is valid. In case that the configuration is not supported, the service provider notifies the device with a negative ACK (NACK), and the interactions completes. Otherwise, it replies back to the device with an ACK message. Subsequently, the user is asked to choose a service type and a channel of interest. Occasionally the selected channel may not be available due to the problem of a content provider. If this happens, the user is expected to select another channel until the selected channel is found to be available. The whole interactions complete after the service provider sends an ACK message.

Then, there is no standard way to describe the interaction logic of this example scenario, in a view point of ETRI's position, user preferences of Web (and Web Services) that can be used in distributed web applications and provide an effective means to flexibly reduce the number of messages exchanged between the client in ubiquitous environment and the Web Service provider, is required within W3C's activities