Dynamic Cache Size Tuning to Shorten Mobile Business Service Roundtrip Time and Turn E-Shoppers into Happy Return Customers

Allan K.Y. Wong, MIEEE and Jackei Wong
Department of Computing, Hong Kong Polytechnic University, Hong Kong SAR
Email: csalwong@comp.polyu.edu.hk

Extended Abstract: The mobile business (m-business) mode has two sequential phases, mobile commerce (m-commerce) and electronic commerce (e-commerce). The m-commerce phase enables sales promotion by e-shops to galvanize potential customers within their short attention spans. Then, sales transactions are handled by the e-commerce phase. In this paper the dynamic cache size tuning technique, MACSC (Model for Adaptive Cache Size Control), is proposed for shortening the m-business service roundtrip time (RTT). This enhances the quality of service (QoS) that creates happy and return customers. The MACSC core is

\[
\text{CacheSize}_{\text{new}} = \text{CacheSize}_{\text{old}} \times \left( \frac{s_r^{\text{current}}}{s_r^{\text{last}}} \right),
\]

which lets the dynamic cache size tuning process maintain the given proxy cache hit ratio on the fly.

The popularity ratio (PR) or \(s_r^{\text{current}}/s_r^{\text{last}}\) is “current statistically measured standard deviation \(s_r^{\text{current}}\) of the ranked objects over the last measurement \(s_r^{\text{last}}\)”. Assuming: a) data retrieval roundtrip time (RTT) between the proxy and a client is \(T_1\) (1st leg); b) the proxy caches the promotion material by e-shops as data objects; c) if the proxy does not have the object, it asks the e-shop(s) over the Internet for it and incurs the \(T_2\) delay (2nd leg); d) the proxy has a given hit ratio of \(\alpha = 0.7\) (70%); and e) \(T_2 = 10 T_1\) due to the Internet DNS (domain name server) delay, then the speedup by caching is

\[
S = (T_1 + T_2) / (0.7 T_1 + 0.3 T_2) = 11/3 = 3.67.
\]

MACSC shortens the m-business RTT by maintaining \(\alpha\) (i.e. for the m-commerce phase).

As a result, it reduces the chance of involving \(T_2\). The PR is based on the Zipf-like behavior, \(y(r) \propto r^{-\beta}\) for \(0 < \beta \leq 1\), where \(r\) is the ranked position of an object. Figure 1 is the log-log plot of \(y(r)\) versus \(r\), and \(r = 0\) for the hottest object with the highest access frequency. MACSC maps \(y(r)\) into a bell curve \(bell(r)\), which is the relative object popularity distribution (PD), on the fly (Figure 2). It uses the two latest successive PD standard deviations (SD) to calculate the current PR. The PD changes shape over time with respect to the shift of user preference towards some data objects. Figure 3 depicts such PD changes (i.e. A, B and C), as reflected by the corresponding \(SD_A\), \(SD_B\) and \(SD_C\).

Test results with the MACSC Java prototype in m-business environments indicate that it indeed maintains the given cache hit ratio (Figure 4). In comparison, the fixed cache-size (FCS) method failed to do so for the same EPA (Environmental Protection Agency (USA)) trace. MACSC shortens the m-business service RTT by reducing the chance of involving the 2nd leg or \(T_2\) over the Internet.

Figure 1. Zipf-like (log-log ) curve

Figure 2. bell(r) = map(y(r))

Figure 3. PD shape changes over time

Figure 4. Changes in hit ratios by FCS and MACSC with the EPA-HTTP data trace