Context Aware QoS and Information Management Services for Proactive Computing

Position Paper

Dr. Joseph Loyall
Michael Atighetchi
Dr. Partha Pal
Aaron Paulos
Matthew Gillen
Jonathan Webb

BBN Technologies
10 Moulton St.
Cambridge, MA 02138

jloyall@bbn.com
matighet@bbn.com
ppal@bbn.com
apaulos@bbn.com
mgillen@bbn.com
jwebb@bbn.com

1 Introduction and Problem Statement

Computing with pervasive, hand-held and mobile devices need an information processing and interaction model that can handle varying bandwidth, can operate seamlessly with varying network and connection qualities, and can provide a consistent level of user experience in situations of increasing load and limited resources, even when detached from centralized servers and enterprise resources. This motivates the need for a number of technical capabilities, including the following:

- An asynchronous interaction paradigm with content-based (not address-based) information exchange and decoupled information providers and consumers.
- Quality of service (QoS) for prioritization of processing and information exchange using constrained and shared resources, based on user preferences, device characteristics, and dynamic conditions.
- Awareness of the user's context, objectives and requirements combined with proactivity to closely match delivered service with the user's purpose, and to adapt to changes in context.

In this short position paper, we describe three technologies that can provide the above capabilities:

- Peer to peer (P2P) Publish-Subscribe-Query (PSQ)-based information management (IM) services
- Dynamic QoS management services for the PSQ IM services
- Context awareness that works with the PSQ and QoS services.

2 Peer to Peer PSQ-based Information Management Services

PSQ-based IM services offer many of the characteristics desired for information interchange by pervasive, mobile, and hand-held devices including support for inherently asynchronous, temporarily disconnected, multi-modal (supporting both data push and pull), and highly dynamic communication. Figure 1 shows the core services of a PSQ-based IM infrastructure. The *Submission service* receives information objects published by applications. The *Broker service* matches registered subscriptions to published information. The *Archive service* inserts published information into an information repository. The *Query service* processes queries and retrieves results from the repository. The *Dissemination service* disseminates the results of brokering to matched subscribers and delivers the results of a query to the requestor. To support the exchange of information between applications on devices without an explicit or static network topology, and without centralized control, global consistency, or transactional behavior in the ad-hoc network of mobile and hand-held devices, the PSQ-based IM service must also be decentralized and P2P, where each device can dynamically pick up a share of the core PSQ services.

A P2P implementation of the core PSQ services enables the decoupled interaction over ad-hoc connectivity. Information publishers and consumers share information without explicitly identifying and connecting to each other, through discovery of peers offering IM services. Information exchange is based on attribute matching in contrast to traditional RPC models, in which clients make calls on services via explicit

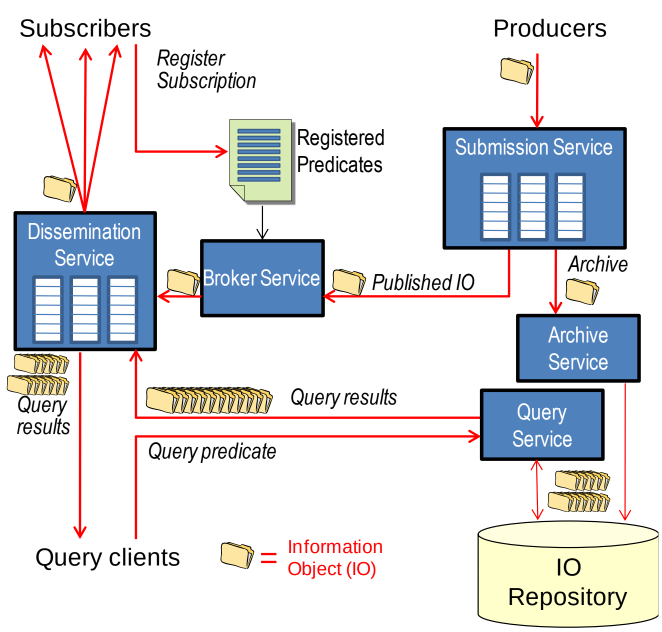

Figure 1. Core PSQ services

references, e.g., a URI, URL, or IP address and port. This has two major aspects:

- Asynchronous communication – Information is published without the need for acknowledgement of receipt by consumers. Likewise, requests are made asynchronously which might be serviced later (when information is published). In each case, the publication and request can be implemented synchronously, but the semantics of the interaction are asynchronous.
- Content-based discovery of information – Since the topology of P2P networks is highly dynamic, with services joining and dropping out, information exchange and collaboration based on addressing is problematic. PSQ services discover information based on content, not location, by matching requests to published information in a decentralized, distributed manner.

3 Dynamic QoS Management Services for P2P IM Services

Pervasive, mobile computing needs QoS management services as part of the middleware infrastructure (Figure 2) lying between the platform and the applications to improve the user's experience and manage constrained and shared resources (including bandwidth, CPU, and memory). This includes deciding which applications and users get to use constrained resources at which times. That is, providing differentiated service based on the importance of applications, users, and operations. It also involves managing quality tradeoffs based on what an application is trying to achieve. For example, whether an application needs more complete information or information faster, or whether an application wants to spend its bandwidth to gain higher resolution or coverage of a larger area.

QoS management for mobile computing is achieved by the following capabilities, provided as part of an overall middleware infrastructure:

- Queuing and prioritization of information to improve network utilization.
- Scheduling of application service processing, thread pools, and thread assignment, matched to the CPU capacity, to optimize end system resources.
- Lightweight proxies that dynamically shape information to fit the available bandwidth and QoS requirements of the application.
- Monitoring of resource availability and usage, including queues, outgoing and incoming bandwidth, and CPU.
- Allocation, control, and exploitation of available network protocols and features.

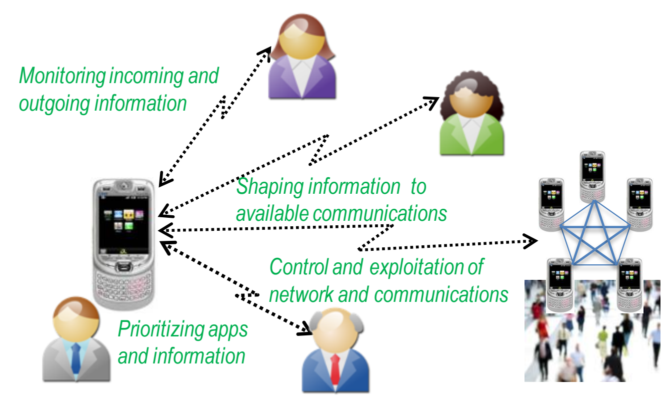

Figure 2. QoS management improves the predictability, control, robustness, and effectiveness of applications and information exchange.

CPU and bandwidth scheduling and prioritization must be matched to the devices' resource capacity, e.g., its ability to simultaneously execute multiple applications. Determining the available bandwidth is a challenge with wireless devices. In theory, the Received Signal Strength Indicator (RSSI) of wireless communications can be used to estimate available bandwidth, but our experience has shown it to be difficult to configure and to result in unreliable estimates. Furthermore, managing bandwidth shared among devices is a challenge. As multiple sources send information to a single target, the resource bottleneck can be the target's network interface card (NIC), affecting available bandwidth for all sources.

Effective QoS management, however, is more than simply allocating resources to applications and information based on priority. It also involves shaping application behavior and information to the capacity of the resources and the needs and desires of the user. It is effectively producing the highest overall user experience within the resources available. Therefore, outgoing and incoming information can be shaped to fit the user's preferences, such as tolerance for delay or loss in bandwidth constrained environments; device characteristics, such as screen size and resolution; and application characteristics, such as whether pan and zoom functions are available. The options for shaping information include rate control, lossless or lossy compression, cropping, scaling, document transformation, fragmentation, and tiling. Tiling and fragmentation each decompose information into pieces that can be separately enqueued, prioritized, and transmitted. Each tile can be received and processed on its own, whereas fragments need to be reassembled to be useful. Examples of tiling include dividing an image into sub-images or dividing a document into pages. Fragmentation incurs less overhead than tiling and is useful to fit transmitted data elements within a Maximum Transmission Unit (MTU) or within a specific time slice, for example. The QoS management capability should be encapsulated as services or composable proxies where possible, so that it can be inserted at the appropriate control points (i.e., Policy Enforcement Points, PEPs) between services, between applications and services, and on the input and output interfaces on devices. We have utilized several openly available software packages for information shaping at various times before, including jMagick, QT, Java Advanced Imaging (JAI), and XSLT processors.

4 Context Awareness for QoS and Information Management

Context Aware information exchange and QoS is an active research area that aims to provide more support for proactivity and usability in mobile environments. Requests for information are automatically supplemented with context information, e.g., the requesting device's location, time, relevant identifying information about the user, device characteristics, and application characteristics.

We have prototyped baseline context awareness in the information brokering services shown in Figure 1. The basic approach is to supplement requests with extra clauses that match on context attributes and supplement published information with additional context metadata. Both of these can be automatically added by the middleware services incorporating contextual information gathered from the device and applications. The information brokering service partitions registered predicates into groups, each of which can be scheduled independently and in the order of importance. The results of matching the context clauses and metadata are used to sort the result set (e.g., using XQuery's "order by" feature) and to invoke the relevant information shaping services prior to information dissemination.

Building upon these baseline context awareness capabilities, we can enhance the IM and QoS service middleware in the following ways to improve the user experience in mobile environments:

- *Incorporate context into requests* to focus on information of imminent concern to the user and able to be displayed on his device, e.g., traffic information is more useful if it highlights congestion that is nearby, along the route being traveled, recently reported, or likely to still present a problem.
- *Prioritize requests and information* that most closely match the user's context, e.g., results requiring immediate action (such as nearby traffic congestion) should be provided more quickly than others.
- *Invoke context-aware information shaping*, resulting in more effective and more efficient information delivery, increasing a user's experience, e.g., if the user is running an application that displays a static map without panning or scrolling functions, imagery destined for that application can be cropped to exactly the size the display can handle and centered on the area of interest. This would increase the user's experience by increasing the speed of response and removing the application-specific unpredictability of how input larger than the display capability is handled.

5 Conclusions

We have presented three areas of active research, all contributing to a vision of proactive computing for pervasive, mobile computing environments. These areas – PSQ information management for P2P information exchange, QoS management for mobile environments, and context awareness – are related and complement one another. The combination of advancements in their development and deployment could result in a major step forward toward the vision of proactive, pervasive computing.