

ExpressQ:

Extending Enterprise Applications to the Field

White Paper

Nettech Systems Inc.

April, 1999



NETTECH
S Y S T E M S

Nettech Systems, Inc.
600 Alexander Road
Princeton, NJ 08540
Tel: 609-734-0300
Fax: 609-734-0346
Web: www.NettechRF.com

© Copyright 1999 Nettech Systems, Inc.

Introduction

Mobile communications present many challenges for software vendors and corporate developers looking to add mobile connectivity to their applications. These challenges are not adequately addressed by data communication techniques used in “wired” environments such as TCP/IP. Application developers creating mobile solutions need to consider several factors that are unique to a mobile environment. Specifically:

- ***Communicating with mobile workers:*** In a mobile environment, coverage is not ubiquitous; users will be moving in and out of coverage. They also use different devices and networks to access the data in the enterprise. For example, the user may want to use dial-up connections to download large messages at the beginning and end of the day, and be able to receive short messages over packet data networks during the day.
- ***“Thin-pipe” networks:*** Wireless networks typically offer much lower bandwidths compared to the Local Area Networks that applications run over in the enterprise, and they also produce higher delays. Also, in many cases, the cost depends on the amount of data sent over the air. Thus, it is crucial that the communication protocol is efficient.
- ***Security of data:*** In mobile solutions, the user may not be physically connected to the enterprise system, and data is sent over the air. Therefore, user authentication and communication encryption issues need to be addressed.
- ***Development Complexity:*** Developers looking to extend enterprise applications to the field are confronted with a myriad of issues such as choosing a mobile platform and network, scalability, integration with back-end systems, and legacy applications.

Middleware products in the market have not been tailored for mobile environments. ExpressQ brings Nettech’s expertise in wireless middleware to provide corporate developers and software vendors an enterprise-strength mobile middleware solution. This White Paper describes ExpressQ, its technology, and its benefits. It also discusses its relationship to other Nettech products.

What is ExpressQ?

ExpressQ is a mobile messaging middleware product that enables corporate developers, as well as software vendors and systems integrators, to quickly extend enterprise applications to a mobile workforce. It addresses challenges of communicating in a mobile environment as follows:

1. It overcomes the challenges of communicating with a roaming workforce by providing the following:

Store-and-forward messaging: ExpressQ utilizes messages queues on the mobile client and server. When a user is out of coverage or not connected (offline), the messages are stored in the message queue and forwarded as soon as the user is connected

“Push” message delivery: ExpressQ stores messages in queues and “pushes” these messages out to clients when available, eliminating the need for a user to “poll” the server to see if any messages are waiting

Network and device independence: ExpressQ utilizes a logical-name addressing approach that allows users to seamlessly receive messages, even when switching between networks. It also allows users access to the network using the device of their choosing. This gives users the flexibility to take advantage of the most cost-effective and convenient communication method available to them.

2. It provides reliable and efficient mobile communications using Nettek's Optimized Wireless Transport Protocol by:

- Dramatically reducing message overhead;
- Compressing data to reduce the amount of traffic sent over the air;
- Providing message packaging, filtering and prioritization capabilities; and
- Automatically adapting to fluctuating coverage conditions.

This results in reduced airtime expenses, extended battery life, and improved performance.

3. It addresses security concerns by:

- Using user names and passwords to authenticate each user; and
- Providing the ability to encrypt data sent over the air

4. ExpressQ provides a development platform that is simple, flexible, and field-proven:
- Supports the broadest range of networks and platforms, giving users maximum flexibility in choosing a network and allowing them to protect their investment in the application as they move from one network/platform to another.
 - Provides consistent APIs across all platforms and networks.
 - Server platform is based on fully scalable Windows NT server.
 - Offers easy integration with back-end systems and applications.
 - Provides a Software Development Kit that supports the most popular development tools and languages.
 - Has been put into production by several organizations, some with several thousand users.

ExpressQ Technology

The high-level system architecture of ExpressQ is shown in Figure 1. ExpressQ consists of client and server software. The ExpressQ Client runs on the mobile computing platform and manages all communication for the application via the ExpressQ API. It supports both wireless and wire-line communications.

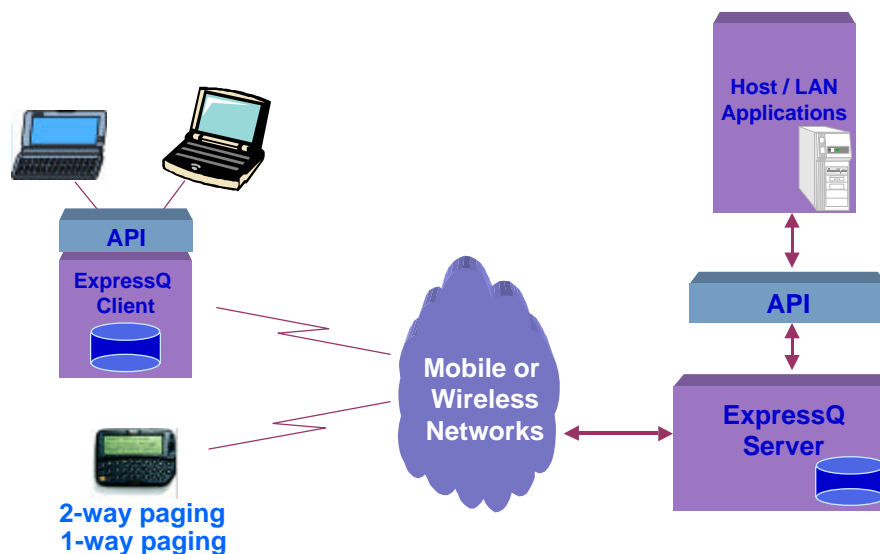


Figure 1. ExpressQ Architecture

The ExpressQ server manages the communication with the clients and routes messages to and from host- or LAN-based applications via the ExpressQ API. On the server, two connectivity options are available: 1) Via an agent using the Messaging API; or 2) Integrated into the server application itself with the Remote API.

A communications session is initiated by the client with the user logging on to the server. The server authenticates the user and acknowledges login acceptance. The client and server are now ready to exchange messages.

Key aspects of ExpressQ technology include:

- Store-and Forward Messaging System
- Intelligent Message Handling
- Network and Device Independence
- Optimized Wireless Transport Protocol
- Mobile Awareness and Control
- Security
- Scalable platform that is easy to integrate
- Multiple APIs to support different environments
- Management Capabilities
- Support for wide array of network and platforms.

These components are described below.

Store-and-Forward Messaging System

ExpressQ has message queues in both the client and the server, which form the heart of the store-and-forward system. These message queues provide temporary storage for inbound and outbound messages. The APIs enable the application to submit and retrieve messages from the message queues.

Message queues provide several benefits to the application including:

- Ability to store messages when a user is not available, to be delivered at a later time.
- Ability to trigger communication based on events using an event-driven architecture.
- Providing the ability to set priority for messages.
- Enabling optimization such as packaging, which is crucial for wireless networks.
- Providing a means for uniform traffic distribution by providing control over how messages of different users are routed.

Intelligent Message Handling

ExpressQ offers a variety of message handling features including:

- **Packaging:** ExpressQ transmits messages in “packages” that contain one or more messages. This reduces the number of acknowledgements required and enables efficient use of the available bandwidth by reducing the number of transport sessions required.
- **Prioritization:** ExpressQ provides a mechanism for setting message priority, allowing higher priority messages to be sent before lower priority ones
- **Delivery/Non-Delivery Notification:** ExpressQ can notify both the client and host application when messages have been successfully delivered. It can also return messages that cannot be delivered for any reason.
- **Message Filtering:** ExpressQ allows the application developer as well or the administrator to set message and network filters to control traffic over wireless networks.
- **Message Lifetimes:** ExpressQ provides the ability to set message lifetimes. If ExpressQ is unable to deliver a message before the lifetime expires, ExpressQ will notify the application so that business rules can be applied for call escalation or other intervention.

Network and Device Independence

ExpressQ uses name-based addressing; it identifies mobile users by name, not by modem ID. When a user logs in using a particular device, the server will know which network to forward a user’s messages to. Thus, messages can be routed independently from the underlying network (wireless or wired) or device. This capability, combined with message filtering, allows the users to roam transparently between networks and to access the system from the network and device that is most convenient and cost-effective for them.

Network types include public packet data networks, circuit-switched networks, packet satellite networks, LANs, and private data networks. ExpressQ also provides interfaces to paging networks, allowing the application to page the user based on rules defined by the application. Device type includes laptops, hand-held computers, palm-size platforms, two-way pagers and one-way pagers.

Optimized Wireless Transport Protocol

ExpressQ uses Nettek's Optimized Wireless Transport protocol, which is a mobile transport layer and connectionless protocol for reliable transmission of messages of any size. Unlike TCP/IP and other protocols designed for wire-based communications, it minimizes overhead and adapts automatically to fluctuating radio conditions. Key capabilities of the mobile transport protocol include:

- Compression to reduce the number of bytes and packets sent over the air.
- Send and receive messages of all sizes over wireless and wire-line networks.
- Adapt automatically to fluctuating coverage conditions.
- Provide communication encryption capabilities.

Mobile Awareness and Control

In order to make the most efficient and intelligent use of the network, it is important for both the user and application to be aware of the current status of the system. This is especially important for wireless networks where network conditions are constantly changing. ExpressQ provides the user and application with status information on the wireless system including:

- Network contact
- Radio signal strength
- Network congestion
- Battery strength.

ExpressQ also provides advanced features specific to a mobile environment including:

- **Modem Control:** ExpressQ allows the application to turn the modem on or off or to be put into sleep mode.
- **Power Management:** ExpressQ provides the ability to recognize suspend and resume commands from the operating system enabling efficient processing.

ExpressQ monitors all information provided by the radio modem and the network regarding the state of the wireless connection and uses that information when making decisions about handling transmissions.

Security

Security is important in wireless environments, as external devices need to access the corporate network and data is being transmitted over the air. ExpressQ addresses these concerns with the following capabilities:

- **Authentication:** ExpressQ users are validated by user names and passwords. This is done at the beginning of each communication session to authenticate the user before any messages can be transmitted.
- **Encryption:** ExpressQ offers the ability to encrypt data transmitted over the air. This is accomplished by designating the encryption key at the server and mobile device. The application performs key management. The encryption type is RC4 using a key that is obtained from the application.

Applications can also be designed to provide additional security above and beyond the level provided by ExpressQ.

ExpressQ Platform

The ExpressQ server platform is based on Windows NT. Windows NT is scalable for deployments from a few hundred users to thousands of users. Windows NT also allows easy integration with existing back-office systems and legacy applications. ExpressQ also takes advantage of NT platform capabilities by allowing it to be run as an NT service. This enables automatic control by the system and provides for efficient operation by providing tighter integration with the operating system.

ExpressQ APIs

ExpressQ provides a number of different APIs, enabling the developer to choose the one that best fits their environment. These APIs provide the application with complete control over all messaging functions including creation, sending, retrieving, extracting and checking the status of a message. In addition, the APIs give the application/developer the ability to control the radio modem, including turning the modem on and off and checking all available network status information. The APIs on the client and server side are identical to simplify application development. Finally, these APIs are consistent across networks and platforms.

ExpressQ also supports a Remote API to enable an application running on a different system (machine) to communicate directly with the ExpressQ server. It provides all the standard messaging functionality offered by the other APIs. The Remote API eliminates the need for an agent, as the application can incorporate ExpressQ function calls directly and communicate with the ExpressQ server via TCP/IP socket connections. This approach is preferred by some developers who are creating new applications and want to make their host application more wireless aware.

Management Capabilities

ExpressQ server provides a GUI that allows the system administrator to oversee the status of the system. It also provides the ability to “ping” users. Large installations can be supported using Nettech’s Mobility Manager server, which is a network management utility.

ExpressQ Networks and Platforms Supported

ExpressQ supports a broad range of networks and platforms. Table 1 summarizes the networks and systems supported by ExpressQ.

Table 1. ExpressQ Network and Platform Support

Network Or System Type	Public Packet Radio Networks					Circuit Switched Networks	LAN	Packet Satellite		Private Packet Radio Networks	
	ARDIS	CDPD	DataTAC 5000	DataTAC 6000	Mobitex	Dial-up	LAN	NORCOM via ARDIS	NORCOM via BSWD	Ericsson EDACS	Motorola Private DataTAC
<i>Mobile</i>	Windows 3.1	√	√	√	√	√	√	√	√		√
	Windows 95/98	√	√	√	√	√	√	√	√	√	√
	Windows NT	√	√	√	√	√	√	√	√	√	√
	DOS	√	√	√	√	√	√	√	√		√
	DPMI	√	√	√	√	√	√	√			√
Server	Windows NT	√	√	√	√	√	√	√	√	√	√

ExpressQ and SmartIP

ExpressQ is part of Nettech's complete product line of mobile middleware products which also includes Smart IP. ExpressQ and Smart IP both utilize the foundation layers for the optimized wireless transport capabilities. ExpressQ is designed as a store-and-forward engine with its own API. Smart IP, on the other hand, is designed to allow native TCP/IP applications (such as browsers, e-mail clients, etc.) to run over any wireless network without modifying the application itself.

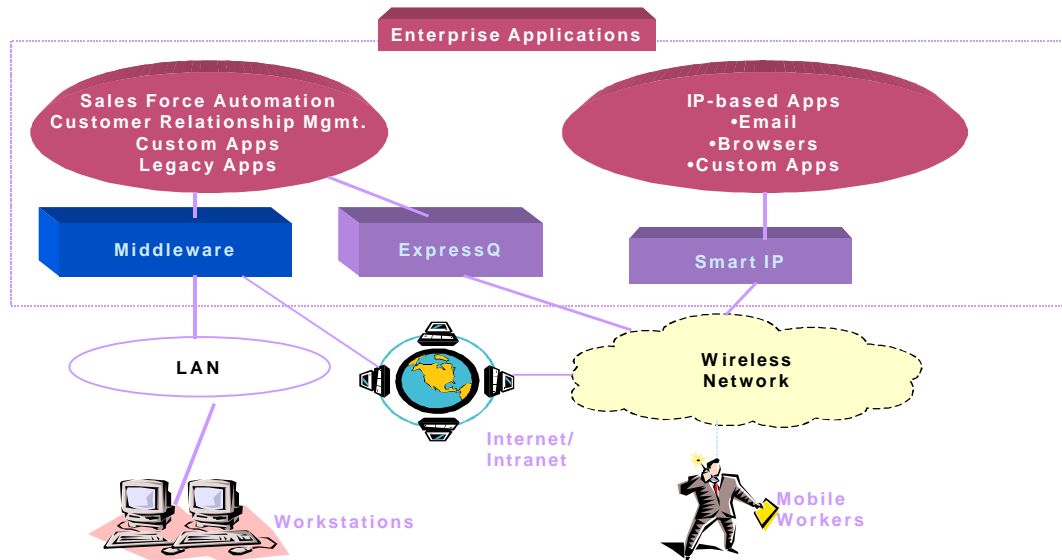


Figure 2: ExpressQ and SmartIP

As shown in Figure 2, different applications may require different products. Customers should choose ExpressQ if:

- Their application requires full messaging middleware including store-and-forward messaging
- Their application requires push messaging and guaranteed message delivery
- They require communications over IP and non-IP networks.

Customers should choose SmartIP if:

- Their application requires IP compatibility out-of-the-box
- They need an optimized transport for interactive applications
- They require a standard Winsock-based interface
- They need a quick and easy way to wireless-enable IP applications for all networks.

ExpressQ Product Evolution

Some additional capabilities planned for ExpressQ in 1999 include:

- **CE Support:** Support for Windows CE as a client platform for ExpressQ.
- **Interactive Pager Support:** Ability to develop applications on the Interactive Pager using the ExpressQ client APIs (ExpressQ currently supports communication with Interactive Paggers).
- **Integration with SmartIP:** Enable ExpressQ to run “on top” of Smart IP allowing native TCP/IP applications and applications written to the ExpressQ API to run on the same machine concurrently. This is shown in Figure 4.
- **Support for Higher-Level Interfaces:** Ability to interface with ExpressQ using COM and/or CORBA interfaces.
- **Connectors to Other Systems:** The capability to access information from other systems such as databases as well as other messaging middleware such as MSMQ.

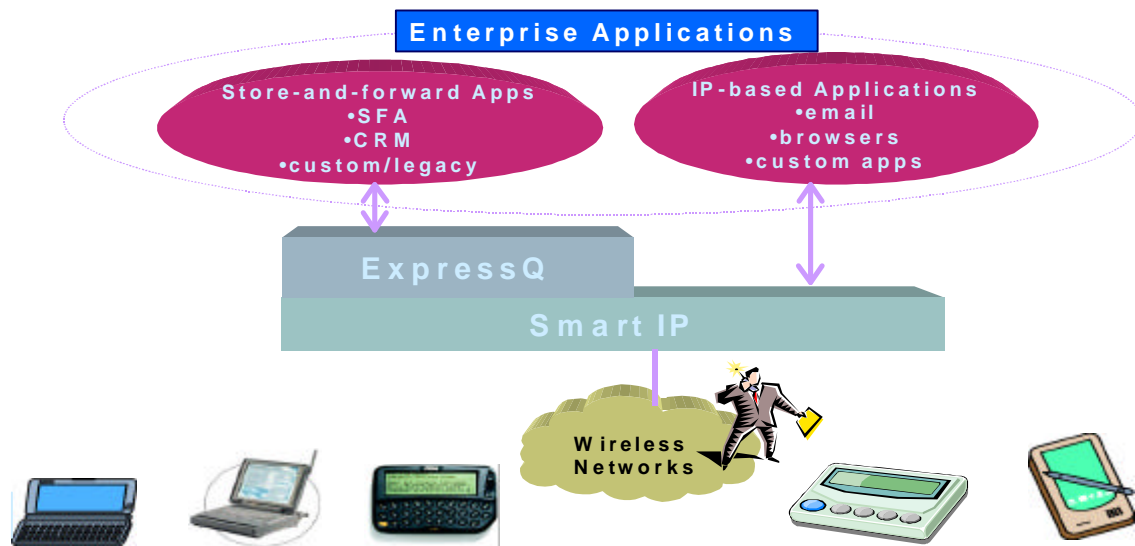


Figure 4. ExpressQ over SmartIP

Summary

ExpressQ is a Mobile Middleware product that can provide corporate developers and software vendors with a fast track to extend enterprise applications to the field by:

1. Reducing time to develop and deploy a mobile solution;
 2. Addressing all mobility issues;
 3. Providing an off-the-shelf communication solution that shields them from the complexities of the mobile computing environment;
 4. Protecting investment in application as you move from one network to another;
 5. Utilizing a fully-scalable platform; and
 6. Reducing their risk with a field-proven solution.
-

Attachment 1

ExpressQ In Action: BST Example

BellSouth Telecommunications (BST) is using ExpressQ to wireless-enable its new TechPlus System. One of the largest deployments of mobile computing technology to date, TechPlus is a field-based workforce computing system that dramatically improves the ability of BellSouth's 15,000 field service technicians to access up-to-the-minute customer service and network status information.

BST provides telecommunications, wireless communications, cable and digital TV, directory advertising and publishing, and Internet and data services to nearly 34 million customers in 19 countries worldwide. BST has wireless-enabled its field dispatch application called TechPlus, using Nettech's ExpressQ. The TechPlus system will be used by the company's 15,000 field service technicians who are responsible for the service and maintenance of the company's network and systems in the nine Southeastern states where BellSouth provides residential and commercial telecommunications services.

The new \$163 million TechPlus solution gives field service technicians both wireline and wireless connectivity to their home bases, providing them with immediate access to existing BellSouth computer systems for network testing, work assignments, maintenance and customer information. During its trial, the system saved technicians an estimated 40 minutes a day in time spent logging onto BellSouth's computer network and receiving and closing out work orders. Additional time-savings can be expected depending on the complexity, number and type of jobs being handled by the technician.

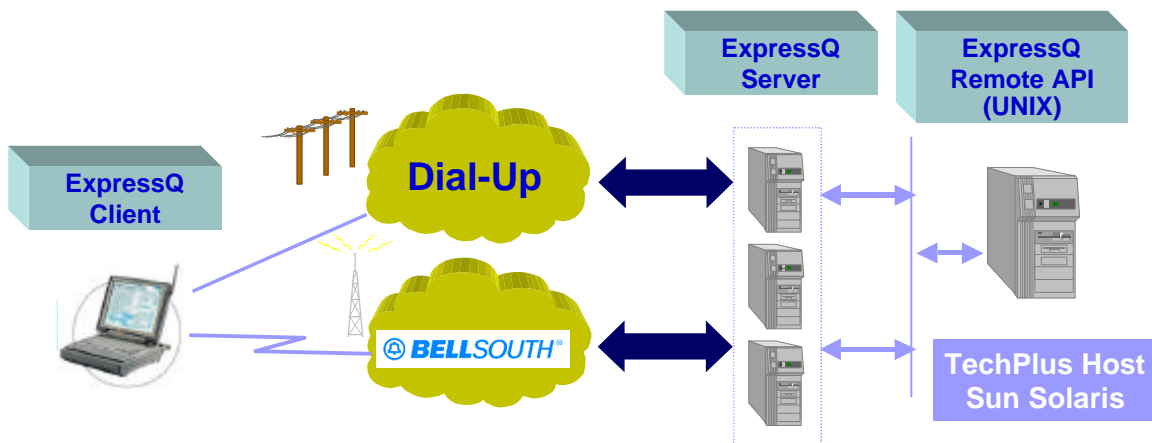


Figure 4. BST Deployment of ExpressQ

BST deployment of ExpressQ is shown in Figure 4. To add wireless connectivity to TechPlus, BellSouth wanted to find an off-the-shelf solution that provided advanced wireless messaging features, supported multiple networks and was field-proven to provide enterprise-strength scalability and reliability. BellSouth chose Nettech's ExpressQ mobile messaging middleware to wireless-enable TechPlus.

Using ExpressQ, BellSouth obtains more than a mere wireless transport; it gains an advanced messaging system that provides a number of additional benefits to the TechPlus System, including:

- Easy integration with the TechPlus application and BellSouth's Sun-based back-end computer platform;
- Enterprise-strength scalability and redundancy;
- Support for automatic switching between the BellSouth Intelligent Wireless networks and circuit-switched cellular or dial up wireline;
- Store and forward messaging for guaranteed message delivery;
- Sophisticated mobile computer power management and over-the-air efficiency; and
- Encryption and authentication to ease security concerns.

Initially, TechPlus will be used primarily for dispatching jobs, receiving and closing out trouble tickets, completing work orders and testing systems for repair of customer lines. BellSouth's fleet of more than 15,000 customer service vans will be equipped with battery charging stations for the Itronix notebook computers, HIGH-gain antennas, and full-sized computer printers. The TechPlus solution utilizes multiple Windows NT Servers running ExpressQ to provide redundancy and a truly scalable enterprise solution. These servers interface with a Sun Solaris system on the back end.

In the future, the system will enable service technicians to accomplish a number of tasks in the field that will further streamline and enhance customer service. These include immediate input and close-out of customer service order requests, electronic capture of customer signatures, processing of credit-card payments for installation of new services or repairs and access to email, the network, construction data, cable planning documents, and other job-relevant information. BellSouth expects to market the TechPlus system to other Regional Bell Operating Companies in the future.