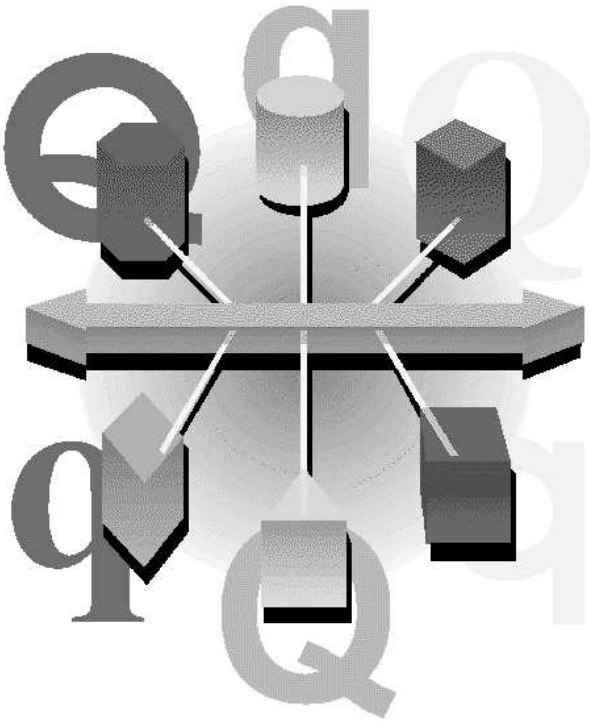
digital

DECmessageQ

Powerful message queuing for distributed application development and integration

Any way you look at it, waiting is a waste of time. Whether you're a developer wrestling with the difficulties of distributed application integration, or a user waiting for a critical application to be developed—no technology tool is worth considering if it doesn't result in increased productivity for the programmer and end user alike. Digital designed DECmessageQ to help save you time by ending the frustration of application development backlog.



DECmessageQ is a powerful development environment that helps you integrate distributed applications quickly across a range of hardware platforms and operating systems. This easy-to-use messaging software enables messages to be sent from one application to another through an advanced queued message bus.

By simplifying the communication interface, DECmessageQ frees your development staff from the intricacies of diverse hardware and networking protocols. As a result, your developers can concentrate their skills on building mission-critical applications that mean success for your business.

An industry leader in message-queuing technology, DECmessageQ is production-proven in mission-critical applications for such industries as:

- Discrete manufacturing
- Process control
- Defense
- Healthcare
- Telecommunications
- Finance and insurance

With DECmessageQ, you can significantly accelerate your move to distributed client/server computing in a multivendor enterprise—saving costs while maximizing the productivity of both your developers and users.

Highlights

- High performance—up to hundreds of non-recoverable messages/second; dozens of recoverable messages/second
- Non-blocking, asynchronous communication for time-independent processing
- Advanced development and management tools
- Broadest platform support in the industry including UNIX(R) and Windows NT(TM) server products
- Microsoft Windows(TM) client product with support for DLLs (Dynamically Linked Libraries), Visual Basic(R) development environments, and mobile/wireless communication systems
- Message recovery on UNIX, Windows NT, Macintosh(R), OS/2(R), and OpenVMS(TM) server platforms and Windows clients
- Interoperability with IBM(R) MVS(R)/CICS(R) and IBM MVS/IMS(R)
- Common Application Programming Interface (API) across all platforms
- Network transport support for TCP/IP, DECnet(TM), and LU6.2

The DECmessageQ Message Queuing Bus: advancing the state-of-the-art

DECmessageQ achieves distributed communications based on a peer-to-peer mechanism called the DECmessageQ Message Queuing Bus. This message bus is a data highway that can be used to transfer messages between processes—whether the message is a request for service across the network or a request for information needed by a distributed application. DECmessageQ provides a set of tools for building message-oriented applications. Systems based on DECmessageQ can be deployed over a variety of software architectures and transports.

Process-to-process communication technology that's fast, asynchronous, and recoverable

As the first company to make message queuing software commercially available, Digital has delivered messaging technology to major private and public organizations worldwide for almost a decade. DECmessageQ helps your developers save time and money in building and integrating modular, distributed applications on systems made by a variety of manufacturers. In short, it reduces time-to-market for your software development efforts. Furthermore, it can preserve your current investments in computing platforms and software applications as you implement future computing strategies.

Guaranteed message delivery for your critical systems

DECmessageQ provides a suite of distributed communication features such as message broadcasting, priority selection, recovery, and guaranteed delivery. Using the software's optional Message Recovery Services, messages can be journaled on local or remote disks prior to their delivery. If a communications line is down, the message is preserved on a local disk and sent again later. If the message has reached a remote disk and the target system goes down, the messages are preserved for delivery after reboot. In either case, the message bus—not the sending application—is responsible for delivering the preserved message.

Message queuing, explained

Message queuing is a method of sending data from one computer program to another in which the data is queued up in computer memory. The message stays in the queue until the target program is ready to receive it—whether it be milliseconds or hours after the message's arrival in the queue.

Queued messaging allows interoperating programs on different computers to exchange data asynchronously. The benefit of message queuing is that the receiving program doesn't have to interrupt processing to wait for incoming data. And in turn, it can pass data destined for another program to a queue and go on processing. The communicating programs can be on the same computer—or on many different computers in an enterprise-wide network.

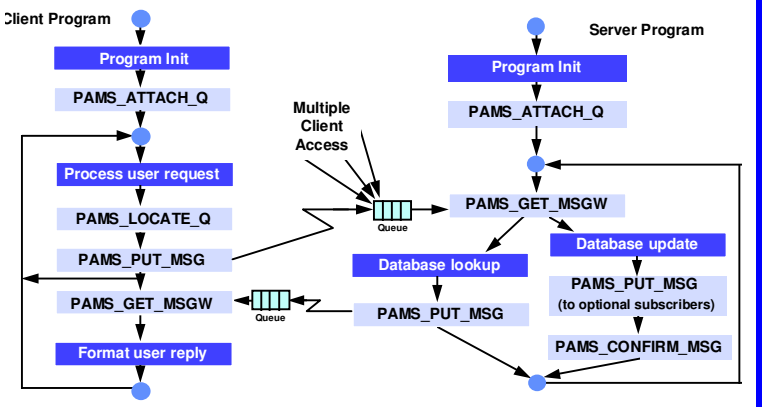
Message queuing lets you integrate new applications with existing applications and systems. Both sets of applications can share the same data by using queued messaging. Because it lets you share data in real time, messaging technology has become especially attractive to manufacturers who want to closely integrate the many steps of their manufacturing operations, and thus achieve more efficient production.

Speed and simplify application development

In addition to the communications capabilities, DECmessageQ also provides application development tools—including an Application Programming Interface (API) and testing and debug facilities. Distributed client/server applications based on DECmessageQ can benefit from the built-in system management utilities. The DECmessageQ monitor helps developers and system managers optimize their system configurations, while the error logging utility helps troubleshoot system problems.

DECmessageQ can be used in conjunction with other popular tools such as Visual Basic and Visual C++, further simplifying the development of applications.

Message Queuing Client/Server Sample Flow



DECmessageQ API call	Function
PAMS_ATTACH_Q	Attach to a message queue
PAMS_LOCATE_Q	Requests the queue address for a specified queue name
PAMS_PUT_MSG	Send a message to a specified target queue
PAMS_GET_MSGW	Wait until a message arrives in a selected queue and retrieve the message.
PAMS_CONFIRM_MSG	Confirms receipt of recoverable message
PAMS_GET_MSG	Retrieve a message from the selected message queue
PAMS_DETACH_Q	Detaches a selected message queue, or all of an application's message queues from the message queueing bus
PAMS_EXIT	Terminates all attachments between an application and the message queueing bus

Features and benefits at-a-glance

Supports synchronous and asynchronous messaging
Application developers can select the messaging style that best serves their application. The familiar synchronous request/response paradigm is easily implemented. The same API can be used to implement one-way messaging—in which the sender does not require a response.

Applications with more time-critical processing requirements can decouple request and response processing, again using the same small set of calls.

Message broadcasting
Message broadcasting allows one message to be sent to several receivers simultaneously. This simplifies application coding because developers do not have to be aware of the number, location, or readiness state of each recipient.

This capability increases the efficiency and performance of a distributed application since a single message can reach several recipients at once. As with other DECmessageQ features, broadcast receivers can be located on any system attached to the message bus.

Message selection
This feature lets application developers use selection filters based on message type, class, sending application, and message data contents. Message selection increases efficiency and performance because messages that are more time-critical are read in priority order.

Message recovery services
Message recovery is a facility that prevents loss of message data. Queues are housed on disk rather than in volatile main memory. If a failure occurs in an application, on a communication line, or on the system where the application is running, data in transit is written to disk and preserved for delivery after the failed elements come online again.

Application development tool kit
This powerful tool kit includes an API, message simulation and capture, and debug capabilities that simplify the development environment. Numerous example programs are also provided as part of every DECmessageQ kit. This single API offers developers a choice of development platforms—easing the development of new applications.

DECmessageQ monitor utility
This utility helps developers and system managers monitor and control the DECmessageQ environment. The error logging utility displays and captures informational, warning, and error messages during processing.

Choice of network transports and client support
DECmessageQ supports TCP/IP, SNA, and DECnet transports—offering a choice of network transport and additional transport interoperability. There is client support for Windows—including Visual Basic, DLLs, and WINsockets.

DECmessageQ optimizes the integration of Windows-based applications. It increases performance, reduces memory requirements, and allows DECmessageQ PC applications to run on today's mobile and portable platforms.

digital

Digital's open, multivendor solution for powerful messaging

Part of Digital's Enterprise Software portfolio, DECmessageQ provides reliable client/server and peer-to-peer communication between applications in a multivendor computing environment.

Use all your favorite platforms

DECmessageQ runs on all major UNIX operating systems, OpenVMS, Windows NT, Macintosh, OS/2, Windows, MS-DOS(R), selected

IBM environments via LU6.2, and on a wide variety of industry-leading hardware, including all three operating systems that run on the high-performance Alpha platform.

The newest version of DECmessageQ adds server support for DEC(R) OSF/1(R), IBM AIX, SunOS(R), Solaris(R), HP-UX(R),Windows NT, OS/2, and Macintosh, a high-performance TCP/IP client for Windows, and recoverable messaging for all platforms—including UNIX, PC DECmessageQ server, and Windows client implementations.

Through DECmessageQ, distributed client/server applications achieve simplicity, interoperability, and portability across the leading industry platforms.

Ordering information for DECmessageQ licenses:

OpenVMS VAX:

QL-GKPA*-AA	OpenVMS VAX DEV
QL-GHVA*-AA	OpenVMS VAX RTO
QL-GKQA*-AA	LU6.2 Services for OpenVMS VAX

OpenVMS AXP:

QL-0HJA*-AA	OpenVMS AXP DEV
QL-0HKA*-AA	OpenVMS AXP RTO
QL-0HLA*-AA	LU6.2 Services for OpenVMS AXP

DEC OSF/1 for AXP:

QL-2W7A*-AA	DEC OSF/1 AXP DEV
QL-2W8A*-AA	DEC OSF/1 AXP RTO

ULTRIX VAX:

QL-MK5A*-AA	ULTRIX/VAX DEV
QL-MKBA*-AA	ULTRIX/VAX RTO

ULTRIX RISC

QL-MK6A*-AA	ULTRIX/RISC DEV
QL-MKCA*-AA	ULTRIX/RISC RTO

HP-UX HP9000:

QL-2W9A*-AA	HP-UX DEV (HP9000-7xx-8xx)
QL-2WAA*-AA	HP-UX RTO (HP9000-7xx-8xx)
QL-MK8A*-AA	HP-UX DEV (HP9000-3xx-4xx)
QL-MKEA*-AA	HP-UX RTO (HP9000-3xx-4xx)

SunOS & Solaris:

QL-MK9A*-AA	SunOS DEV
QL-MKFA*-AA	SunOS RTO
QL-2WDA*-AA	SOLARIS DEV
QL-2WEA*-AA	SOLARIS RTO

IBM AIX RS6000:

QL-2WBA*-AA	AIX DEV
QL-2WCA*-AA	AIX RTO

Other UNIX:

QL-MK7A*-AA	88open DEV
QL-MKDA*-AA	88open RTO

PC Servers:

QL-110AY-AA	Windows NT DEV (AXP/Intel)
QL-113AY-AA	Windows NT RTO (AXP/Intel)
QL-2WFAW-AA	Macintosh DEV
QL-2WGAW-AA	Macintosh RTO

(Coming soon OS/2 DEV and OS/2 RTO)

PC Clients:

QL-MK4AW-AA	Macintosh Client
QL-GHRAW-AA	MS-DOS/MS-Windows Client
QL-GHSAW-AA	OS/2 Client
QM-2WHAA-AA	MS Windows Client (Intel)

Note: DEV = Development, RTO = Run Time Only

Want more information?

To learn more about DECmessageQ or any other Digital product or service, please contact your authorized Digital distributor or local Digital sales office.

Digital conducts its business in a manner that conserves the environment and protects the safety and health of its employees, customers, and the community.

Digital believes the information in this publication is accurate as of its publication date; such information is subject to change without notice. Digital is not responsible for any inadvertent errors.

AXP, DEC, DECmessageQ, DECnet, Digital, the DIGITAL logo, OpenVMS, and VAX are trademarks of Digital Equipment Corporation.

Macintosh is a registered trademark of Apple Computer, Inc. IBM, AIX, and OS/2 are registered trademarks of International Business Machines Corporation. HP-UX is a registered trademark of Hewlett- Packard Company. Windows, NT, Windows NT, MS-DOS and Visual Basic are registered trademarks of Microsoft Corporation. Sun, Solaris, and SunOS are registered trademarks of Sun Microsystems, Inc. OSF/1 is a registered trademark of Open Software Foundation, Inc. Intel is a trademark of Intel Corporation. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd.

Part #: EC-F3400-48.

Copyright 1994 Digital Equipment Corporation. All Rights Reserved