Software Product Description

PRODUCT NAME: ObjectBroker for MVS, Version 2.5A SPD 56.01.00

DESCRIPTION

ObjectBroker for MVS® Version 2.5A provides application developers and system integrators with the development tools and runtime environments to integrate applications and services distributed across MVS environments.

ObjectBroker Version 2.5A is Digital's compliant implementation of the Object Management Group's (OMG®) Common Object Request Broker Architecture (CORBA™) Version 1.2 specification.

ObjectBroker simplifies distributed application development by providing platform-independent, client-server, programming interfaces. It offers an object-oriented (O-O) approach to dynamic linking of independently developed applications and services. ObjectBroker allows application processes to transparently invoke, control, and interact with each other across multiple platforms

Features

- A single high-level, standards-based application programming interface (API) across many hetrogeneos platforms.
- Compliant with Object Management Group's CORBA V1.2 specification, protecting users' software development investments.
- O-O abstraction provides insulation of clients and servers from changes in network topology or application code through the use of implementation-independent brokering services.

ObjectBroker for MVS, Version 2.5A capabilities:

- The Interface Definition Language (IDL) for describing classes and messages.
- Support for User Defined Types (UDT).

- Generation of client-side stubs for operations defined on one or more interfaces:
  - Static interfaces use code skeletons (stubs) containing routines that cannot be changed. Stubs provide a linear interface and reduce the complexity of the method resolution.
  - Dynamic interfaces use routines that the client defines and builds as it is running. The dynamic interface provides client with more flexibility to use deferred synchronous operations, different method maps, and new interfaces.
- Generation of IDL skeletons (dispatchers), registration routines, and method routine stubs for servers.
- Support for the Dynamic Invocation Interface (DII) routines.

Interface Repository

The interface repository is a storage container for the definitions of interfaces, operations, and attributes which a client can use to make requests of the Object Request Broker (ORB). The interface repository can also contain method maps, which describe the criteria by which to select implementations. The interface definitions in the Interface Repository are written in CORBA IDL, which is described in the CORBA V1.2 specification and in ObjectBroker documentation.

Context Object

A context object contains information about a user's preferences, which are taken into account during the method resolution process, and about property values, which are propagated to the server side during an invocation request. In addition, applications can store profile information in the context object that is not used as method resolution criteria.

Agent

The agent is the entity provided as part of ObjectBroker V2.5A that performs services on behalf of a user in the context of ObjectBroker. An agent must be present on any host that supports applications that provide services using ObjectBroker.

Security Registry

The security registry contains information about a user's authorization to access specific implementations (method servers) and methods. The implementations and methods referenced by the security database must be registered in the Implementation Registry. Information about the agent is the only exception to this policy because the agent and its methods are entities that are well known to the ORB.

A user's authorization information is maintained in the form of an access control list (ACL). This list contains the information about which methods of an implementation a user is allowed to access. Therefore to start a server, the user must have access to the StartServer method of the agent; this is the default. Failure to have this access restricts a user from starting any new servers.

Along with authorization information for users, the security database also contains information that describes the list of remote users that are allowed to access ObjectBroker on the current host. This information allows system administrators to specify a mapping between a user on a remote host and a local user account. This allows for remote users to access server applications on the local hosts that are not available to most users.

ObjectBroker Version 2.5A security capabilities include:

Access control lists specify a list of users that have access to specific servers and to methods within a server. Server startup control give the system administrator the ability to control whether servers can be started on a given host, regardless of the authentication setting. Additionally, ACL's can be used to control whether a user is authorized to start a server on a host.

Advertisement Registry

Advertisements are the primary source of information to both a client application and the ORB Core. A client application can retrieve initial objects from the advertisement registry.

ObjectBroker's advertisement registry stores information about servers, where to find them, and what they provide. Advertisements are created when an implementation is installed somewhere on the network. The generated advertisement can be distributed at any time to remote advertisement registries.

Implementation registry

The implementation registry contains information about object implementations that have been installed on this host. At a minimum, the implementation registry contains the unique identifier of the implementation, the operating system specific command for starting the server process that contains the implementation code, and the list of attributes to be defined as environment variables when the process that executes the implementation is started.

Configuration registry

The configuration registry contains information about the configuration of ObjectBroker on a given node. Configuration information stored in the configuration registry includes the list of available transports, the authentication package to be used, and the location of files that contain the various implementations for the transport, security, and repository replaceable components. The configuration registry includes the list of locations for shared resources such as context objects, advertisements, and repositories.

Digital Extensions to the CORBA specification are not required by an application but provide additional capabilities. These extensions include:

Languages

- The Implementation Language (IML) describes methods and method servers that constitute the implementation of an object.
- The Method Mapping Language (MML) maps operations to the methods that implement them.

The ObjectBroker Comand Utility Interface for MVS provides a new, fully supported, panel driven set of utilities that help system integrators manage the CORBA environment and programmers develop new distributed applications. The utility provides interfaces that:

- View and manipulate context objects
- View and manipulate repositories
- View and manipulate remote servers
- Generate code for programmers
- Set up the environment (security, proxies, transport)

The user interface of each utility is based on CORBA concepts. The utility's interface is native to the platform on which it is executing (Motif, Windows, Macintosh, ISPF panels).

Tranports

ObjectBroker supports IBM'sTCP/IP.

Network Test Tool

The network test tool provides the ability to determine whether the network is properly configured for use with ObjectBroker. A user can specify the target host to test against and the amount of data to be sent. This tool also provides detailed error messages when failures occur.

Installation

ObjectBroker for MVS is installed using a simple procedure. The installation manual that accompanies the software, describes individual steps and pre-requisites for installation.

HARDWARE REQUIREMENTS

Disk Space Requirements

ObjectBroker for MVS requires 350 tracks

This value refers to the minimum disk space required on system/user DASD. This size is an approximate; actual size may vary depending on the user's system environment, configuration, and software options.

SOFTWARE REQUIREMENTS

Operating System

- MVS 4.3 with JES 4.3 or higher
- TSO 2.4, ISPF + ISPF/PDF 3.5
- SAS/C® C Compiler version 5.5.0

Transports

- VTAM 4.2, TCP/IP 2.2.1

GROWTH CONSIDERATIONS

The minimum hardware/software requirements for any future version of this product may be different from the requirements for the current version.

DISTRIBUTION MEDIA

- 6250 BPI magtape
- 3480 tape cartridge

ORDERING INFORMATION

Development

Platform	License Order Number	Software Services Order Number
Group 20	Q2-4LRAX-AA	Q2-4LRAX-L9
Group 25	Q2-4LVAX-AA	Q2-4LVAX-L9
Group 30	Q2-4LWAX-AA	Q2-4LWAX-L9
Group 35	Q2-4LXAX-AA	Q2-4LXAX-L9
Group 40	Q2-4M0AX-AA	Q2-4M0AX-L9
Group 50	Q2-4M2AX-AA	Q2-4M2AX-L9
Group 60	Q2-4M4AX-AA	Q2-4M4AX-L9

Platform	License Order Number	Software Services Order Number
Group 70	Q2-4M5AX-AA	Q2-4M5AX-L9
Group 80	Q2-4M7AX-AA	Q2-4M7AX-L9
Group 90	Q2-4M9AX-AA	Q2-4M9AX-L9
Group 100	Q2-4MBAX-AA	Q2-4MBAX-L9

Runtime

Platform	License Order Number	Software Services Order Number
Group 20	Q2-4LSAX-AA	Q2-4LSAX-L9
Group 25	Q2-4MCAX-AA	Q2-4MCAX-L9
Group 30	Q2-4MDAX-AA	Q2-4MDAX-L9
Group 35	Q2-4MEAX-AA	Q2-4MEAX-L9
Group 40	Q2-4MFAX-AA	Q2-4MFAX-L9
Group 50	Q2-4MGAX-AA	Q2-4MGAX-L9
Group 60	Q2-4MHAX-AA	Q2-4MHAX-L9
Group 70	Q2-4MJAX-AA	Q2-4MJAX-L9
Group 80	Q2-4MKAX-AA	Q2-4MKAX-L9
Group 90	Q2-4MLAX-AA	Q2-4MLAX-L9
Group 100	Q2-4MMAX-AA	Q2-4MMAX-L9

ObjectBroker for MVS Software Media & Documentation Kits

ObjectBroker for MVS Development
Q2-4LRAA-HM Magtape
Q2-4LRAA-HP 3480 Cartridge Tape

ObjectBroker for MVS Run-Time Only
Q2-4LSAA-HM Magtape
Q2-4LSAA-HP 3480 Cartridge Tape

ObjectBroker for MVS Development Documentation Only
Q2-4LRAA-GZ

ObjectBroker for MVS Run-Time Documentation Only
Q2-4LSAA-GZ

For additional information on available licenses, services, and media, refer to the appropriate price book.

SOFTWARE LICENSING

This software is only furnished only under a license. For more information about Digital's licensing terms and policies, contact your local Digital office.

SOFTWARE PRODUCT SERVICES

A variety of service options are available from Digital. For more information, contact your local Digital office.

SOFTWARE WARRANTY

A 90-day warranty for this software product is provided by Digital with the purchase of a license.

The above information is valid at time of release. Please contact your local Digital office for the most up-to-date information.