Middleware Integration

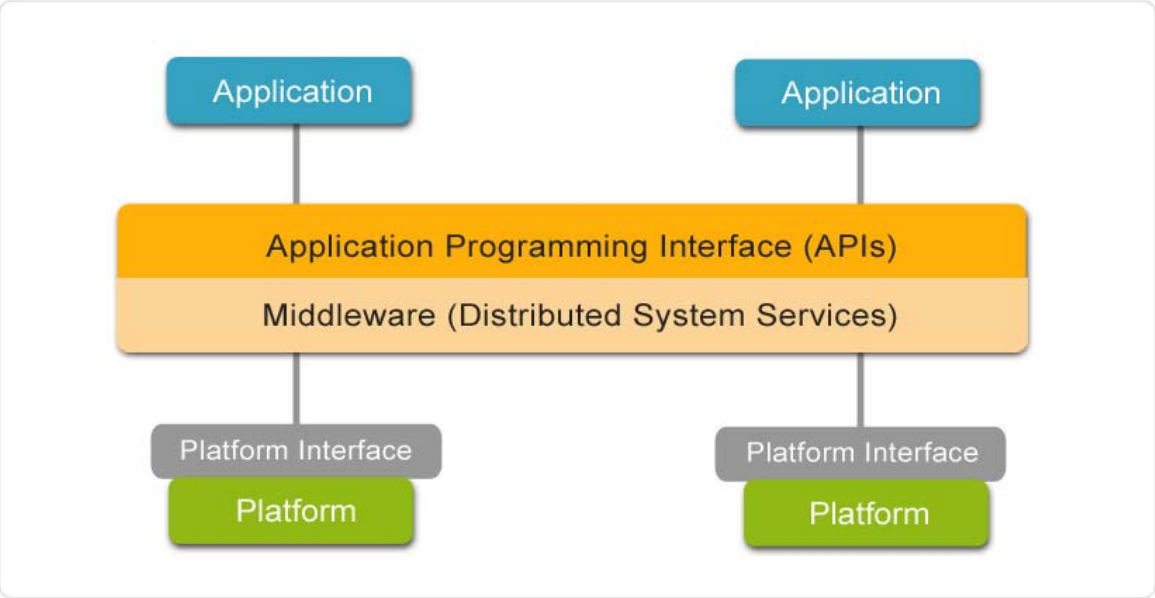
Middleware is a general term for any programs that link or mediate between two separate programs. Middleware is critical to transferring mainframe applications to client/server applications and to providing for communication across heterogeneous platforms. This technology has developed during the 1990s to provide for interoperability in support of the move to client/server architectures.

The main purpose of middleware services is to help solve many application connectivity and interoperability problems. Middleware services provide a more functional set of Application Programming Interfaces (APIs) than the operating system and network services to allow an application to 1) locate across the network, providing interaction with another application or service, 2) be independent from network services, 3) be reliable and available, and 4) scale up in capacity without losing function.

Middleware can take on the following different forms:

FORMS	DESCRIPTION
Transaction processing (TP) monitors	An application server permits a large number of users to execute applications using a minimum of system resources. An application server can be extended to allow coordinated transactions to be invoked from applications executed by the application server. This transaction coordination is generally known as a Transaction Processing (TP) monitor. A TP monitor works in conjunction with an application server.
Remote Procedure Call (RPC)	RPC is a powerful technique for constructing distributed, client-server based applications. It is based on extending the notion of conventional or local procedure calling, so that the called procedure need not exist in the same address space as the calling procedure. The two processes may be on the same system, or they may be on different systems with a network connecting them. By using RPC, programmers of distributed applications avoid the details of the interface with the network. The transport independence of RPC isolates the application from the physical and logical elements of the data communications mechanism and allows the application to use a variety of transports.
Message-Oriented Middleware (MOM)	MOM is a client/server infrastructure that increases the interoperability, portability, and flexibility of an application by allowing the application to be distributed over multiple heterogeneous platforms. It reduces the complexity of developing applications that span multiple operating systems and network protocols by insulating the application developer from the details of the various operating system and network interfaces- Application Programming Interfaces (APIs) that extend across diverse platforms and networks are typically provided by the MOM
Object Request Brokers (ORB)	ORB is middleware that uses the CORBA specification. The Object Request Broker or ORB takes care of all of the details involved in routing a request from client to object, and routing the response to its destination. The ORB is also the custodian of the Interface Repository (abbreviated variously IR or IFR), an OMG-standardized distributed database containing OMG IDL interface definitions.

The most widely-publicized middleware initiatives are the Open Software Foundation's Distributed Computing Environment (DCE), Object Management Group's Common Objest Request Broker Architecture (CORBA), and Microsoft's Component Object Model (COM/DCOM). A widespread application of middleware is to allow programs written for access to a particular database to access other databases (such as VSAM, IMS, Oracle, or Adabas) without any custom coding.



Timewell provides various middleware solutions that simply business integration. We can assist your projects at any level. We can design an architecture based on the product you are trying to deliver, and then develop a prototype to prove that the architecture works and scales. We are working on the standard connections allowing your organizations to quickly and easily control technologies such as EJB, J2EE, CORBA, IBM's Websphere, Microsoft BizTalk Server, XML and Java.