Microsoft Integration Technologies
FOR APPLICATION INTEGRATION

Author: Mohammad Yusuf Dandeli

1. Abstract
This document details on scenarios for Microsoft integration solutions. Equipped with the knowledge of the options available, one can simplify the process of choosing the best Microsoft technologies for application integration.

2. Why Application Integration?
Many enterprises around the world attempt to satisfy new integration functionality requirements. These are originated by issues such as

- The use of legacy systems and heterogeneous technologies,
- High communication development through the Internet that increases the business between enterprise and enterprise consumer,
- Dynamics of business groups such as joint ventures, Merger & Acquisitions and takeovers

To carry out these requirements, enterprises have chosen to reconsider and eventually reprogram their computer applications, to implement other enterprise packages or to integrate disparate information systems. Factors such as the high cost in the development of new applications, strong trust on legacy systems, and the need to quickly integrate disparate information systems have made the decision of implementing application integration more attractive.

To address this broad set of problems, Microsoft has created several different integration technologies, each targeting a particular group of scenarios. Together, these technologies provide a comprehensive, unified, and complete integration solution. Microsoft's integration technologies can be grouped into several categories as described in this document. Each of these technologies has its own distinct role to play in integrating applications.

3. Integrating Applications directly
Every application has some core logic which distinguishes it from application data. The simplest way to connect application logic is to connect directly one part of an application to another using some kind of remote procedure call (RPC). .NET Framework applications have following options for doing RPC.

3.1. ASP .NET Web Services (ASMX)
ASP.NET Web Services (ASMX) lets Windows applications communicate directly with applications running on Windows or other operating systems via SOAP, the foundation protocol for web services.

3.2. .NET Remoting
.NET Remoting lets Windows applications communicate directly with other Windows applications using the traditional distributed objects approach.
3.3. Enterprise Services

Enterprise Services lets Windows applications communicate directly with other Windows applications, letting those applications use distributed transactions, object lifetime management, and other functions.

3.4. Windows Communication Foundation (WCF)

WCF is an extension to the .NET Framework for building service-oriented applications. It is a secure, reliable, and scalable messaging platform for the .NET Framework 3.0 or above applications. WCF uses IPC (named pipes), TCP, HTTP and MSMQ protocols to transfer messages.

4. Integrating Applications through Queues

Using queues as a direct communication between parts of applications is not a best solution. This style of integration relies on the presence of one or more queues between the sender and receiver, with applications sending messages to and receiving messages from these queues. The benefits of this method are:

- Queued communication lets applications interact in a flexible, adaptable way.
- One major benefit of this approach is that the receiving application need not be ready to read a message from the sender at the time that message is sent. In fact, the receiver might not even be running when the message is sent. Instead, messages wait in a queue, usually stored on disk, until the receiver is ready to process them.

4.1. Microsoft Message Queuing

Microsoft Message Queuing (MSMQ) is the built-in technology in Windows for application-to-application communication using queued messaging.

The primary integration scenarios for MSMQ are:

- When asynchronous communication is required between two or more Windows applications
- When the sender and receiver might not be running at the same time
- When message-level logging is required

4.2. SQL Service Broker

SQL Server Broker (SSB) is a communication technology provided as part of SQL Server 2005. SSB provides the infrastructure support for message queuing, as MSMQ does. It provides an alternate for standalone queuing infrastructure as MSMQ does, SSB provides queued communication using SQL Server 2005.
The primary integration scenarios for SSB are:

- Connecting logic built as stored procedures in one or more separate instances of SQL Server 2005
- Connecting logic built as a .NET Framework application using SQL Server 2005 with a stored procedure in the same or another instance of SQL Server 2005

Any organization that relies heavily on SQL Server 2005 for building applications, especially when those applications are implemented as stored procedures, will likely use SSB for communication. As it is part of SQL Server, SSB allows users to have a single product to install, configure, and monitor, together with a single approach to failover, for both a DBMS and a queuing technology.

4.3. WCF and Queued Communication

WCF’s channel architecture allows it to send SOAP messages over diverse protocols. For direct communication with non-Windows systems, WCF will typically send SOAP over HTTP. For queued communication between Windows applications, WCF will also be able to send messages over MSMQ and SSB.

5. Integrating with Applications and Data on IBM Systems

Microsoft Host Integration Server (HIS) is a set of technologies focused on connecting to applications and data on IBM mainframes and mid-range systems.

5.1. Microsoft Host Integration Server

Many enterprises have substantial investments in IBM mainframe and midrange systems. When new applications are written on Windows, integrating with existing applications and data on these older systems is often essential. Yet doing this can be challenging, since these environments support applications and store data in several different ways.

Effectively linking Windows software to these existing IBM systems requires a variety of approaches. HIS contains components that address these diverse requirements. Using various parts of the product, Windows software can access applications and data on IBM zSeries mainframes running z/OS, along with applications and data on IBM iSeries mid-range systems running OS/400. HIS 2004 also includes an MSMQ-MQSeries Bridge, allowing queued messaging between MSMQ and IBM's WebSphere MQ.
The primary integration scenarios for HIS are:

- Connecting Windows systems to IBM zSeries mainframes and iSeries midrange systems using Systems Network Architecture (SNA) and other IBM communication technologies, including SNA over TCP/IP.
- Integrating Windows security with IBM mainframe or midrange security systems, including IBM's Resource Access Control Facility (RACF) and Computer Associates' ACF2 and Top Secret.
- Accessing existing CICS and IMS applications, either directly from .NET Framework applications using HIS's Transaction Integrator or via Web services.
- Creating Windows applications that access data stored on zSeries and iSeries systems, including VSAM data and relational data stored in DB2.
- Connecting MSMQ to IBM's WebSphere MQ, allowing messages to be transferred between these two message queuing technologies.

6. Integrating Applications through a Broker

Rather than integrating applications directly, it sometimes makes more sense to connect them via a broker. A broker is software that sits between the applications being integrated, interacting with all of them. By providing a common connection point, brokers avoid the complexity that can arise when several applications are connected directly to one another. Brokers can provide a range of integration services, including transformations between different message formats and support for diverse communication technologies. A broker can also act as a platform for its own application logic, providing the intelligence to control a business process. For Windows, connecting applications through a broker would imply using BizTalk Server 2006.
Microsoft BizTalk Server 2006

BizTalk Server 2006 is an integration and business process platform and is Microsoft's solution for brokered application-to-application integration. BizTalk Server 2006 sits in the middle of a group of applications. BizTalk Server 2006 provides adapters for various communication mechanisms, including MSMQ, EDI, HL7, and many more. Following are the some of integration capabilities provided by BizTalk Server 2006:

- The ability to graphically define orchestrations, logic that interacts with applications on other systems to drive an integrated process, together with runtime services for orchestrations, such as state management and support for long-running transactions
- Graphical definition of XML schemas for messages, along with the ability to define transformations between incoming and outgoing messages that use those schemas
- Business-to-business (B2B) integration features, including support for Electronic Data Interchange (EDI), RosettaNet, HL7, and other standard interchange formats

The primary integration scenarios for BizTalk Server 2006 are:

- Creating brokered application-to-application message-based integration, especially when data mapping and support for diverse communication mechanisms is required
- Implementing integration processes, including long-running processes that take hours, days, or weeks to complete, and processes with complex business rules
- Addressing B2B integration, including situations with many trading partner interactions and those that require industry standards such as RosettaNet and HL7
• Creating business processes that give information workers real-time visibility into an integrated process

7. Integrating Data

Integrating data means moving and manipulating passive information of the application which is application data.

7.1. SQL Server Integration Services

SQL Server Integration Services (SSIS) provides tools for combining data from diverse data sources into a SQL Server 2005 and above database. SSIS is the extract, transform, and load (ETL) service for SQL Server 2005 and above.

The primary integration scenarios for SSIS are:

• Combining information from a group of operational databases into a data warehouse. Along with powerful support for data transformations, SSIS provides graphical tools for defining the ETL process, fuzzy logic for data cleansing, error handling, and other features to make integration of diverse data easier.

• Transferring data from one DBMS to one or more other DBMSs. Because SSIS supports heterogeneous data sources, the products involved might or might not be SQL Server 2005.

• Loading data into SQL Server databases from flat files, spreadsheets, and other diverse data sources.

7.2. SQL Server Replication

SQL Server Replication allows replicating data across two or more SQL Server databases. It's often useful to have a copy of the same data in multiple databases, and then have that data automatically kept in sync. For example, letting applications running on a group of web servers spread their read requests across a group of identical databases, each on its own machine, can improve the application's scalability and availability. For this arrangement to work, all updates...
must go to a single database instance, and then be propagated to the read-only copies. SQL Server Replication is designed for situations like this.

The primary integration scenarios for SQL Server Replication are:

- Replicating data between tables in one or more SQL Server instances. Those instances might be running on servers, clients, or even mobile devices that are only occasionally connected. Rather than copying entire tables, SQL Server Replication replicates incremental row-level changes, letting updates propagate at near real-time speeds.
- Using SQL Server as a source for data that is replicated to IBM and Oracle databases.
- Using Oracle as a source for data that is replicated to SQL Server, IBM, and Oracle databases.

8. Conclusion

Different integration requirements require different solutions, and it's important to use the right tool for the job. To address the diversity of applications and data that must connect, Microsoft has produced a range of integration products and technologies. These solutions sometimes overlap, and so more than one choice might be applicable to a given situation. While these ambiguous cases are relatively uncommon, the most straightforward way to make a decision is by examining the fundamental scenarios for each integration technology.

9. Appendix

<table>
<thead>
<tr>
<th>Integration Style</th>
<th>Technology</th>
<th>Primary Integration Scenarios</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrating Applications directly</td>
<td>ASP.NET Web Services (ASMX)</td>
<td>Connecting .NET applications with Windows and non-Windows applications via SOAP</td>
</tr>
<tr>
<td></td>
<td>.NET Remoting</td>
<td>Connecting .NET applications with other .NET applications via distributed objects</td>
</tr>
<tr>
<td></td>
<td>Enterprise Services</td>
<td>Connecting .NET applications with other .NET applications that use distributed transactions, object lifetime management, etc.</td>
</tr>
<tr>
<td></td>
<td>WCF</td>
<td>Connecting .NET applications with Windows</td>
</tr>
</tbody>
</table>
### Integrating Applications through Queues

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Message Queuing</td>
<td>Connecting Windows applications with other Windows applications using queued messaging</td>
</tr>
<tr>
<td>SQL Service Broker</td>
<td>Connecting SQL Server 2005 applications with other SQL Server 2005 applications using queued messaging</td>
</tr>
<tr>
<td>WCF</td>
<td>Connecting .NET applications with other .NET applications using queued messaging (via MSMQ and/or SSB)</td>
</tr>
</tbody>
</table>

### Integrating with Applications and Data on IBM Systems

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Host Integration Server</td>
<td>Connecting Windows applications with IBM zSeries and iSeries applications and data</td>
</tr>
<tr>
<td></td>
<td>Connecting MSMQ with IBM WebSphere MQ</td>
</tr>
</tbody>
</table>

### Integrating applications through a broker

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft BizTalk Server 2006</td>
<td>Connecting Windows applications and non-Windows applications using diverse protocols</td>
</tr>
<tr>
<td></td>
<td>Translating between different message formats</td>
</tr>
<tr>
<td></td>
<td>Controlling business processes with graphically-defined orchestrations</td>
</tr>
<tr>
<td></td>
<td>Connecting with business partners using industry standards, such as RosettaNet and HL7</td>
</tr>
<tr>
<td></td>
<td>Providing business process services, such as Business Activity Monitoring and a Business Rules Engine</td>
</tr>
</tbody>
</table>

### Integrating data

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL Server Integration Services</td>
<td>Combining and transforming data from diverse sources into SQL Server 2005 data</td>
</tr>
<tr>
<td>SQL Server Replication</td>
<td>Synchronizing SQL Server data with copies of that data in other instances of SQL Server, Oracle, or DB2</td>
</tr>
</tbody>
</table>

### References

1. [http://technet.microsoft.com](http://technet.microsoft.com)
About CSC

The mission of CSC is to be a global leader in providing technology-enabled business solutions and services.

With the broadest range of capabilities, CSC offers clients the solutions they need to manage complexity, focus on core businesses, collaborate with partners and clients, and improve operations.

CSC makes a special point of understanding its clients and provides experts with real-world experience to work with them. CSC is vendor-independent, delivering solutions that best meet each client’s unique requirements.

For 50 years, clients in industries and governments worldwide have trusted CSC with their business process and information systems outsourcing, systems integration and consulting needs. The company trades on the New York Stock Exchange under the symbol “CSC.”