P2PIE: A NEW ENTERPRISE APPLICATION INTEGRATION SOLUTION

Jörgen Hanson[1], Jens Ohlsson[2], Nevzat Ertan[1], Paul Johannesson[2], Stefan Wernmo[3], Ole Schjødt-Osmo[4], Shengnan Han[2], Tomas Hedlund[2]

[1]Sandvik AB, Stockholm, Sweden
{jorgen.hanson, nevzat.ertan}@sandvik.com
[2]Department of Computer and Systems Sciences, Stockholm University
Kista, Sweden
{jeoh, pajo, shengnan}@dsv.su.se
tohed86@gmail.com
[3] Husqvarna AB
Stockholm, Sweden
swernmo@gmail.com
[4]Yara International
Oslo, Norway
ole.schjodt-osmo@gmail.com

Abstract. In order to provide unified and consistent data and functionality, enterprises need to make all their applications work in an integrated manner. However, integration has become more difficult to handle when enterprises extend their business globally, more applications are implemented for supporting business activities, and more computing paradigms are adopted, e.g., cloud, social, and mobile, just to name a few. The state-of-the-art integration solutions have limitations to address the difficulty. In this paper, point to point integration engine-P2PIE, a new integration solution is presented and discussed. The solution combines distributed execution of integration with centralized management console, which innovate the fundamental integration infrastructures of point to point paradigm, and message systems. P2PIE is developed and tested by Sandvik over the decade. The solution has addressed Sandvik's challenges in enterprise application integration. Agility, cost reduction and reliability of this solution are proved.

Keywords: enterprise application integration, point to point, message systems, integration solution

1 Introduction

In order to provide unified and consistent data and functionality, enterprises need to make all their applications work in an integrated manner. However, integration has been a difficult task to handle for several reasons. First, integration is highly depended on specific applications to be integrated, which involves integrating applications of various kinds, such as custom-built applications based on C++/C#, Java/J2EE, packaged applications, such as CRM or ERP applications, and legacy applications, for instances, mainframe CICS or IMS. Second, these applications are usually dispersed geographically when enterprises expend their business globally. Third, these applications are mostly operated on various platforms. Fourth, when enterprises increase cooperation and build up partnerships with other companies, there is a demand for integrating applications across enterprises.

Over the last one and a half decade, we have witnessed significant evolution in enterprise application integration (EAI). Integration paradigms have developed from simple file/data sharing, messaging system, message broker to more complete integration solution, which has a centralized point of control for increasing the flexibility in changing and managing integration [1, 2]. Message brokers (hub-and .spoke) and enterprise service buses (ESB) are the two state-of-the-art integration solutions that are adopted by enterprises [3]. The goal of the paradigm evolution has been to increase reliability, effectiveness, and reduce the time it takes to change and reconfigure integration solutions, as well lower the cost of integration.

The state-of-the-art integration solutions have limitations. This resides partly on the integration logic for solving an integration problem that is highly distributed in its nature by centralized software and hardware [4]. Furthermore, demands for integration and complexity of integration are intensified. According to [5], organizations application portfolios are increasing in complexity by deploying more applications overall as well as more application functionalities in the cloud, on mobile devices and in conjunction with external business partners. B2B growth, cloud adaptation (SaaS), M&A, and consolidation activities are also foreseen to result in increased integration demands. Furthermore, the demand is accelerated when enterprises are transforming into digital business. Integrating big data and any things from human, machine and applications is emerging as the greatest challenge that enterprises have to address. A new enterprise application integration solution is emerged and developed to handle the new challenges.

Sandvik, a large international company producing metal tools, headquartered in Sweden, has managed its current and future integration challenges by developing this new integration solution based on the business demands for agility, cost reduction, and reliability. The solution is named Point to Point Integration Engine (P2PIE). The P2PIE solution has been developed independently but some of its features, in particular the point to point and message mechanisms, are similar to those state-of-the-art solutions that suggested by [6, 7, 8].

The paper is organized as follows. In section 2, related work for enterprise application integration is briefly discussed. The case of Sandvik and its motivation for developing P2PIE are described in section 3. Section 4, P2PIE solutions are discussed, including its architecture, components, as well the gained values of implementing P2PIE. The paper ends with discussions and conclusion. Abbreviations used in this paper are given at appendix A.

2 Related work

Enterprise application integration (EAI) is aimed to make applications be able to communicate with each other. The paradigms for EAI have evolved over time, from point to point solution, to message brokers (hub-and-spoke), and to enterprise services bus (ESB)

The integration logic of a point to point integration solution is to connect two applications which need to integrate with each other [4]. This type of communication has two ends, and does not need to go through a central server to handle the integration. In a fully integrated enterprise where all applications are connected with each other, with N applications, there will be $\frac{N(N-1)}{2}$ connections. The advantage of a point to point solution is that it communicate directly between the points. The disadvantages of this solution are that every point's integration solution is uniquely built and that the points as a whole have no overlaying monitoring ability [4].

The integration logic of message brokers (hub-and-spoke) is built upon message systems infrastructure [1]. This solution can transform, route and queue messages through a central messaging hub [9, 10]. Most of enterprise applications have various data format, which raise problems in integration. Message broker can solve the problem by transforming data from different applications into message. The message is routed through a central messaging hub that enable integration among these applications [9]. The advantage of a message broker is that it offers a centralized integration solution where the integration logic and error management are collected in one place [1]. The disadvantage of the solutions is that all communication goes through the central hub, as the solution grows the more pressure will be put on the hub.

An ESB has the same integration logic as message broker. However, ESB transforms data and route messages by using services for achieving integration goals [2]. The services needed are implemented separately and therefore allows applying more functionality without disturbing the rest of the solution [9, 10]. The advantage of an ESB is that it is flexible, allowing it to grow with the enterprise's need. The disadvantage of this solution is that it is complex, as the services are managed separately.

3 Integration problems at Sandvik

Sandvik, recognized as a world leader in tools and application solutions for the metal-cutting industry, grew through many acquisitions during the period 1995-2005. A multi-brand strategy and a strong decentralization culture resulted in a legacy consisting of hundreds of business critical applications on different heterogeneous platforms and systems. These applications were communicating through various integration solutions /platforms from different vendors. Which resulted in little synergies between the main business sectors. Furthermore, most of the integration tasks were completed within each business sector by using "hard coupled" systems integration logic. Consequently, integration projects can't be controlled within budget and completed on time. Moreover, the clear roles/mandates between central/local organizations can't be specified, as well the central monitoring and governance of the enterprise application integration can't be done.

Sandvik followed an approach based on Microsoft BizTalk, an ESB solution. Still, due to the increasingly higher volumes of data and the synchronous connections to and from Sandvik's back end system, there were however still challenges to obtain a highly reliable environment. An intermediate solution was then to add an asynchronous queuing approach by IBM Webshere MQ. In this way some of the problems could be solved but also new ones were introduced, basically depending on the fact; by adding this MQ component between the sources system / receiving system and BizTalk, the control over the complete integration chain from the source system to the receiving system get lost. All these experiences called for development of a new solution that is robust, reliable, flexible, easily scalable and cost efficient.

In 2005, Sandvik decided to create a common artefact for connecting its backend systems to MQ, and this artefact would be governed and monitored centrally. While doing so, it was discovered that it would be fairly easy to make this artefact adapter based. This meant that a component was built that was distributed to all servers where it was needed, which had the capability of transferring data between any two endpoints. It also has the capability of remote management, and the source or destination endpoints could be changed from the central Management Console just by changing configurations. P2PIE solution, thus, is developed for solving EAI problems at Sandvik.

4 Point to Point Integration Engine -P2PIE

The P2PIE solution retains all benefits in a hub and spoke architecture by keeping management central (including configuration, control and monitoring), but decentralizing the Integration Engine in order to optimize efficiency and reduce complexity.

The key components of P2PIE solution include integration engine, central management console, and a central database. The architecture of P2PIE is shown in Figure 1.

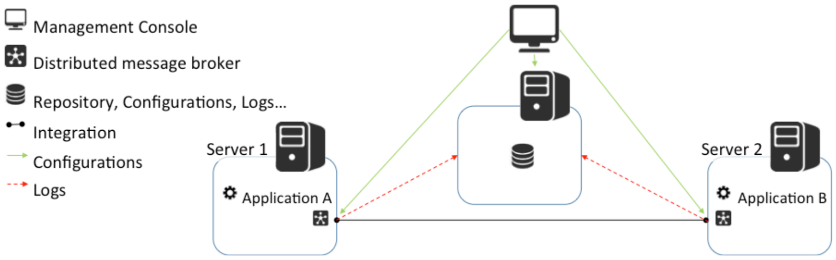


Fig. 1. Point-to-Point Integration Engine Architecture

The P2PIE is installed on the same physical servers that serves as endpoints in an integration scenario. On a conceptual level, this can be seen as a big distributed "virtual broker", fully operated from a central management console. The individual distributed Integration Engines can act as both sources and destinations in integration scenarios.

The P2PIE in Sandvik handles the integration of applications and uses a MQ to transport messages between points. The integration line of sight is the physical reach of the integration components (integration engines). P2PIE connects business applications as endpoints directly to each other (endpoint to endpoint), thus delivering 100% line of sight. This differs from how a Broker/ESB sees the communication and integration between two applications. A Broker/ESB is put in the middle and only sees what happens with the messages up to the MQ. The line of sight is therefore limited to that point and there is no control over what happens between the application and the MQ. This is not delivering transparency into the business applications since the Broker/ESB can't see when applications reads or puts something on the queue. This difference is illustrated in Figure 2.

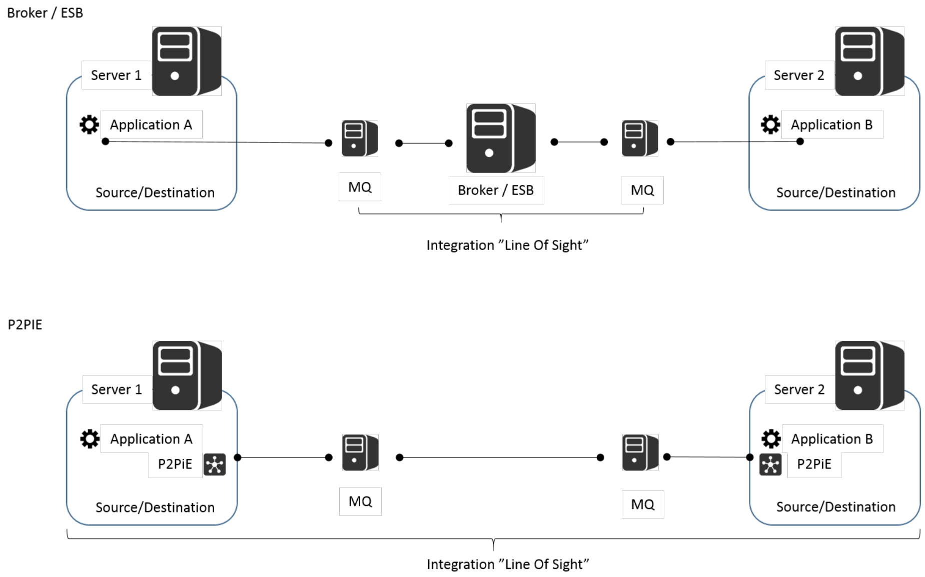


Fig. 2. Integration "Line of Sight": a comparison between Broker/ESB solution and P2PIE

4.1 Integration Engine

An "Integration Engine", which shortly can be described as an adapter based component that is installed on all servers with software applications that need to communicate and integrate (Figure 3). The logic is thus executed in the engine and the engine itself is distributed by installing a software "integration instance", corresponding to one service, on each server. Depending on integration scenario the server may have several instances. The source-destination connections are set up by DLL-based plug-ins in the integration engine. One plug-in for the source application but an arbitrary number of plug-ins for the needed destination applications. Also other types of plug-ins than source-destination plug-ins can be used, e.g. pre-processors or post-processors for specific tasks. e.g. XSLT-transforming, advanced routing, or any type of enhancement to the inbound message. All plugins are stored in the central Repository and are distributed on a need-to-have basis.

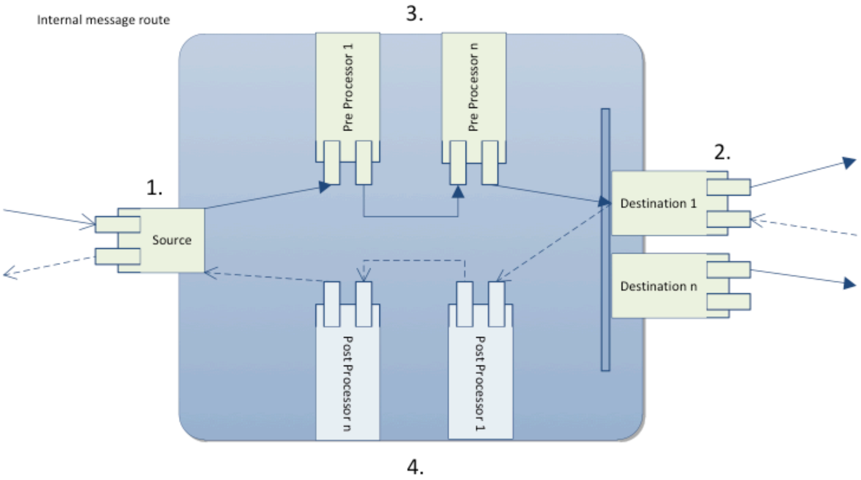


Fig. 3. Distributed Integration Engine within the P2PIE solution.
Note: 1. Source, 2. Destination 1..n, 3. Pre-processor 1..n, 4. Post-processor 1..n

P2PIE also provides an API so that an external application can send, or retrieve messages via a P2PIE instance. If an exception occurs, it is translated into a WMI event and can thus be monitored by SCOM. Each integration point keeps track of each event, when it starts, stops, or throws an exception and distributes the information to the central repository in nearly real time. The system monitors each step in the integration chain and alarms during deviation from the predetermined thresholds. This means that an error in the integration-chain can be quickly found and addressed to the individuals with the right skills for the task.

P2PIE is a distributed integration. However, it possesses the possibility to do service oriented integration. Since service oriented integration doesn't necessarily need to use a broker or ESB, which works well in P2PIE solution. The scalability and the resource capability of each server are not an issue in P2PIE. Sandvik monitors each server to measure its utilization. The evidence shows that capability deficiency is rarely happened in servers.

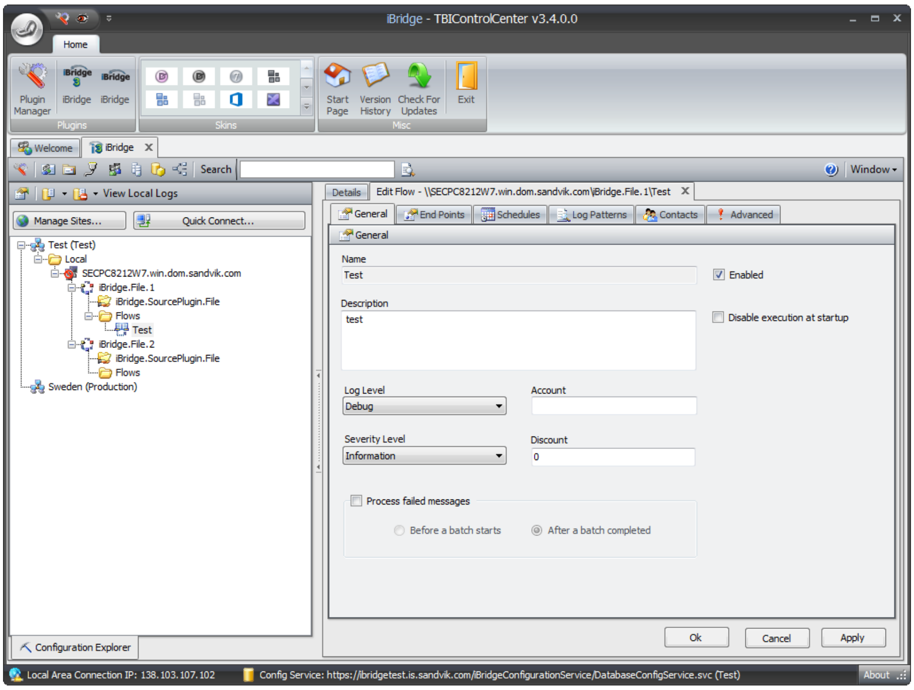
4.2 Central Management Console

A central "Management Console", where all integration engines are governed and monitored remotely and centrally. The GUI of the central management console of P2PIE is shown in Figure 4. The central console configure the Integration Engine, and thus integrations are created. All configurations made are stored into the central databases and to all Integration Engines that taking part in integrations. From the central management console, every execution related to an integration is controlled, including deployment, start/stop, monitoring, alerts, notifications and exception handling.

4.3 Database

P2PIE has a central database running on SQL server. This database does not have any run-time dependencies to the distributed Integration Engines. The database saves a copy of all configurations done, a repository for all plug-ins including their history, and central log storage.

- Configuration
  This is where a central copy of all configurations are persisted
- Repository
  The repository is the storage area for all plugins that are used in the P2PIE solution. The history of plugins is also kept so that a rollback of configuration is possible.
- Logs
  Whenever an integration is executed, i.e. one business application sends and/or receives information from another business application, an information-log is created and stored in the central log database.



iBridge - TBIControlCenter v3.4.0.0

Test (Test)
Local
SECPC8212W7.win.dom.sandvik.com
iBridge.File.1
iBridge.SourcePlugin.File
Flows
Test
iBridge.File.2
iBridge.SourcePlugin.File
Flows
Sweden (Production)

Edit Flow - \\SECPC8212W7.win.dom.sandvik.com\iBridge.File.1\Test

General

Name: Test

☑ Enabled

Description: test

☐ Disable execution at startup

Log Level: Debug

Account:

Severity Level: Information

Discount: 0

☐ Process failed messages

○ Before a batch starts ● After a batch completed

Local Area Connection IP: 138.103.107.102

Config Service: https://ibridgetest.is.sandvik.com/iBridgeConfigurationService/DatabaseConfigService.svc (Test)

Fig. 4. The GUI of the central management console

4.4 Gained value of P2PIE at Sandvik

P2PIE has matured and grown internally at Sandvik. It now handles all integrations where BizTalk is not involved and manages more than ten times the daily load of the BizTalk environments. This is by far the largest integration solution at Sandvik.

P2PIE is installed on more than 250 servers worldwide at Sandvik and manages more than 2,500 integration points, handling more than 500 000 transactions/day. The solution makes integration through configuration possible on the following systems and platforms: BizTalk, .NET, MS SQL Server, SAP, IBM MQSeries, Lotus Notes, Java, AS/400 and IBM mainframe.

The development and implementation of the P2PIE solution has brought significant positive effects and business value to Sandvik. The business demands for enterprise application integration on agility, reliability, and cost reduction have achieved:

- Improved agility: P2PIE reduces the change and implementation times more than 30%. This is possible both as integrations are more direct between two endpoints, which leading to reduced complexity. Moreover, integration instances can be reused for configuration, which taking less time than coding again in creating an integration.
- Increased reliability: P2PIE improves monitoring, enhances identification of exceptions, and handles exceptions by better and more efficient tools. The reliability of the integration related to the data flow and the content consistency have increased significantly. In 2012 the P2PIE solution handled more than 120 000 000 messages with an average of 7 exceptions per 100 000 messages. Volume evolution and reliability over time of P2PIE at Sandvik is shown in Figure 5.

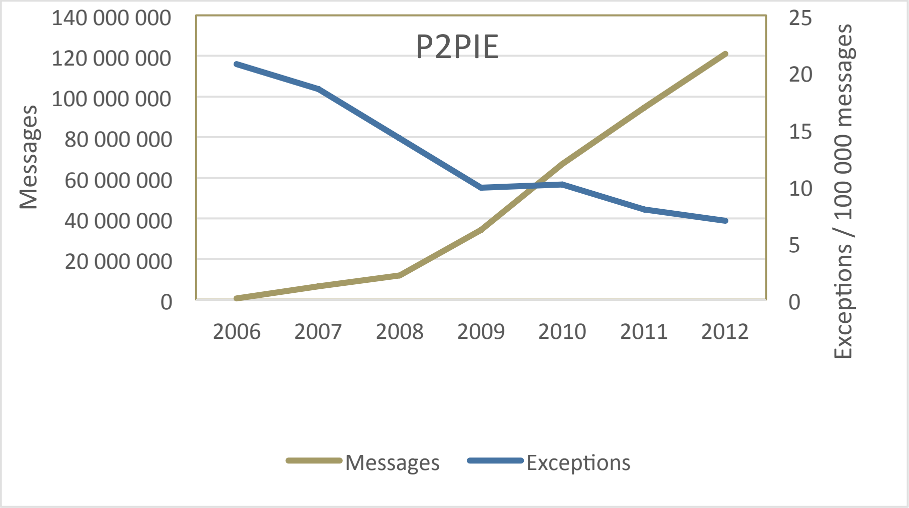


Fig. 5. P2PIE at Sandvik - Volume evolution and reliability over time

- Reduced cost: P2PIE reduces cost in two aspects. Firstly, it reduces cost in setting up integrations in a new system development project. The integration cost in a new system development project, e.g., ERP system, was estimated at 35% of the total project budget in 2002. This was reduced to 25% 2007, and 15% 2012. Secondly, P2PIE reduces the annual running cost for operations and maintenance of the current legacy integration environment. Due to the large reduction of vendor integration platforms/suites and cost avoidance of several BizTalk instances, the percentage of the integration related cost in systems operations & maintenance was reduced from roughly 21% 2002 to 3% 2007, and even lower at 1,5% at 2012. These savings were generated from reduced maintenance costs, less hardware cost, and more reuse and configuration of implemented integrations.

5 Discussion and conclusion

In this paper, P2PIE, a new integration solution developed by Sandvik, is presented and discussed. The implementation of P2PIE has achieved business values for Sandvik that the company gains the capability to response changes efficiently and to pursue business opportunities effectively. P2PIE solution is aimed to lower cost, improve agility and reliability.

P2PIE integration solution is better aligned with the nature of managing a data flow across an IT landscape in enterprises. The logical flow schemes and control mechanisms are kept at a central console, while the physical point to point integration engine is implemented by taking into account the distributed nature of integration. Thereby it pursues efficient execution of the real time operational data flow. By doing so, data are allowed to flow the shortest and most cost effective way between the actual applications and databases.

Despite its simplicity and alignment with the nature of integration (with requirements for central control but distributed execution of data flow), the P2PIE solution makes significant contribution to EAI research and practice. Although the P2PIE integration solution is built upon the well- known and proven paradigm of point to point integration paradigm and message systems, the solution innovates EAI by using ***distributed and decentralized logic*** rather than the state-of-the-art *centralized* approach. This underlines the originality and innovativeness of P2PIE integration solution. The values achieved by implementing P2PIE at Sandvik over the years also bring strong empirical evidence for demonstrating the capability of the solution in tackling integration challenges. The novelty of the P2PIE integration paradigm is its distributed nature, where business applications send and receive data directly without passing a centralized hub, which also delivers the capability of full transparency of integration flows.

In the recent years, the need to manage a growing demand for integration is increased and intensified. Since many companies have to integrate applications used by different partners on B2B value networks, to migrating to emerging computing paradigm, e.g. cloud, mobile, social, big internet of things, cyber physical systems, just to name a few. Companies have also to meet demand for services to be available everywhere on every new devices. P2PIE is an innovative solution that can be used for addressing these integration challenges. Future research is to demonstrate and evaluate the P2PIE solution in additional instantiations both at large enterprises as well as medium sized companies.

A. Appendix: Abbreviations

A2A - Application to Application
B2B – Business to Business
CICS - Customer Information Control System (mainly run on IBM mainframe)
CRM – Customer relationship management
DLL - Dynamic-Link Library
ESB – Enterprise Service Bus
EAI – Enterprise Application Integration
ERP- Enterprise Resource Planning
GUI – Graphical User Interface
HTTP - Hypertext Transfer Protocol
IMS - IBM Information Management System
M&A – Mergers and Acquisitions
MQ - Message Queueing
P2PIE – Point to Point Integration Engine
R&D – Research and Development
SaaS – Software as a service
SCOM - System Center Operations Manager
SOAP - Simple Object Access Protocol or Symbolic Optimal Assembly Program
SQL - Structured Query Language
XML - Extensible Markup Language
XSLT - XML (Extensible Markup Language) Stylesheet Language for Transformations
WMI - Windows Management Interface

References

1. Ferreira, D.R.: Enterprise Systems Integration: A Process-Oriented Approach. Springer, Heidelberg (2013)
2. Roshen, W.: SOA-based enterprise integration: A step-by-step guide to services-based application integration. McGraw-Hill, New York (2009)
3. Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web services: Concepts, Architectures and Application. Springer, Heidelberg (2004)
4. Linthicum, D.S.: Enterprise application integration (Addison-Wesley information technology series). Addison-Wesley, Reading (2000)

5. Lheureux, B. J., Pezzini, M., Thompson, J., Altman, R., Sholler, D., Schulte, W. R., Malinverno, P., Knipp, E: Predicts 2013: Application Integration. Gartner (2012)
6. Kupsch, F. and Werth, D.: Integrating Business Processes with Peer-to-Peer Technology. In: Konstantas, D., Bourrières, J.P., Léonard, M., Boudjlida, N. (eds.) Interoperability of Enterprise Software and Applications, pp. 277–288. Springer, London (2006)
7. Lin, A., Maheshwari, P.: Agent-Based Middleware for Web Service Dynamic Integration on Peer-to-Peer Networks. In: Zhang, S., Jarvis, R. (eds.) AI 2005: Advances in Artificial Intelligence. LNCS, vol. 3809, pp. 405–414. Springer, Heidelberg (2005)
8. Pallickara, S. and Fox, G.: NaradaBrokering: A Distributed Middleware Framework and Architecture for Enabling Durable Peer-to-Peer Grids. In: Endler, M. and Schmidt, D. (eds.) Middleware 2003, LNCS, vol. 2672, pp. 41–61. Springer, Heidelberg (2003)
9. Gold-Bernstein, B., Ruh, W.A.: Enterprise integration: The essential guide to integration solutions. Addison-Wesley, Boston (2005)
10. Brahm, G.: ESB vs EMB. http://esb-nitin.blogspot.se/2009_07_01_archive.html (2009)