UNIT 5

ENTERPRISE APPLICATION INTEGRATION AND SUPPLY CHAIN VISIBILITY

5.0 ENTERPRISE APPLICATION INTEGRATION

Enterprise application integration (EAI) is a practice that enables internal and external collaboration by interconnecting people and information systems. Enterprises tend to evolve tailored functional applications to meet specific business needs. It's been a common practice to implement ERP applications for integrating internal business functions and processes. The dynamic business needs prompt enterprises to consider outside-in approach for implementing enterprise applications for enabling supplier and customer relationships. Over a period of time, the investments on enterprise information systems are realized with returns in terms of expected business benefits. Since information systems become backbone for enterprise wide information processing, they are considered to be valuable assets that require maintenance. As the enterprise transforms with changing business conditions, enterprise systems deployed by multiple vendors tend to evolve and create islands of information. The reasons being issues related to information technology platforms, data format and applications incompatibility. Enterprise application integration is a practice that enables integration of such disparate systems to create seamless enterprise wide information flow, also known as *system of information systems*. It provides integration approaches to derive most value out of existing information systems investments. EAI includes message acceptance, transformation, translation, routing, message delivery and business process management.

Drivers of EAI

The prime reasons for EAI are as follows:

1. **Inorganic growth:** Enterprises grow organically through increase in customer base &profitability and inorganically through mergers and acquisitions (M&A). In the cases of M&A, enterprises require rapid integration of disparate business processes between two enterprises. Faster business process integration enables the new enterprise to gain strategic control.
2. **Electronic business:** Execution of e-business strategy requires usage of internet technologies to improve collaboration with suppliers, customers and partners. In order to create value chain and supply chain over the internet, it requires integration of various enterprise systems. While core enterprise systems work as back bone for information processing and exchange, the internet enabled systems improves human-system interactions. Enterprise application integration allows integration of core enterprise systems with internet enabled applications to create effective e-business model.
3. **Statutory laws and industry regulations:** Enforcement of statutory laws by governments and industry regulations by authorized agencies force enterprises to comply. Design and development of better compliance processes related to taxation, customs and excise, intellectual property, national security…requires greater degree of integration with enterprise information systems. The varied compliance across countries makes it complex for global enterprises manage compliance issues.
4. **Business process management / improvement:** Managing and improving internal and external business process efficiency requires innovative applications deployment. New

applications are to be integrated with existing applications to derive full potential of process innovation or complete alignment of systems with business goals.

5. **Enterprise resource planning systems:** The increased deployment of effective ERP systems requires integration with existing legacy functional applications.
6. **Demand and supply chain management:** The practice of demand and supply chain management requires integration of information systems across trade partners. Shared and linked business processes among supply chain partners to service end customer needs makes information integration a pre-requisite for better collaboration.
7. **Real time enterprise or zero latency enterprise:** To be successful in dynamic markets, enterprise has to change business rules in real time. Ability to change business rules in real time enables enterprise to exploit market opportunities. Business rules are statements describing the policies and practices of an organization. They represent operational policies and structures (explicit or declared business rules) or system policies and architectures (implicit rules). *For example, for example, departmental activities, job descriptions, pay scales, procurement procedures, and stock replenishment practices are explicit business rules that are part of business processes. Implicit rules are data based models related to information systems.* With business rules spread across islands of information systems, enterprises are constrained to change them according to business needs. Enterprise application integration allows enterprise to abstract core business rules for improvedagility.

The above points indicate that enterprise application take two forms (refer figure 1):

1. Intra enterprise application integration
2. Inter-enterprise application integration

The *intra enterprise application integration* is of two forms: horizontal and vertical. The horizontal EAI refers to integration of systems supporting various functions. For example, supply chain management practice requires integration of multiple processes across functions inside an enterprise as a pre-requisite. It involves scenarios of integration between different functional applications components supporting multiple functional processes or ERP system components with other functional applications components. The vertical EAI refers to integration of systems across different levels for administrative control. For example, integration of process control systems, numerical control systems and manufacturing execution system (MES) with ERP allows flow of data from process level to planning level in the enterprise. It involves scenarios of capturing, transforming and loading relevant data from multiple data bases spread across disparate systems to create decision support systems.

The *inter-enterprise application integration* refers to integration of systems between two enterprises. For example, in supply chain integration scenario of type III partnership, focal enterprise, the logistics service provider and the financial institution should share transaction data in a common format among three or more disparate systems. EDI standards are being used for communication between multiple enterprises and are being replaced by relatively simple extended markup language (XML) based applications.

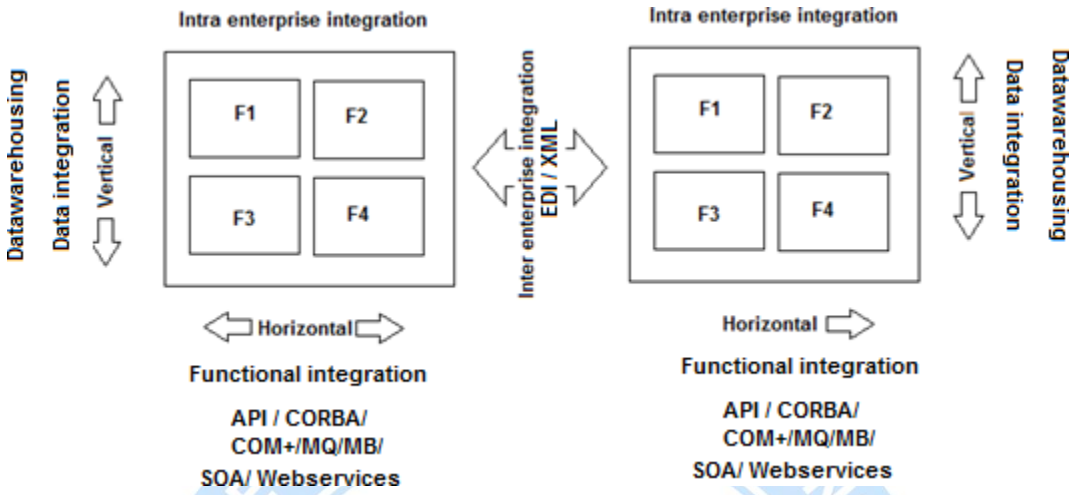

Figure 1: Intra & Inter enterprise application integration

Middleware

Middleware is a type of software that provides the connectivity between network services, applications and business processes in a distributed computing system. Figure 2 illustrates the role of middleware in enterprise application integration.

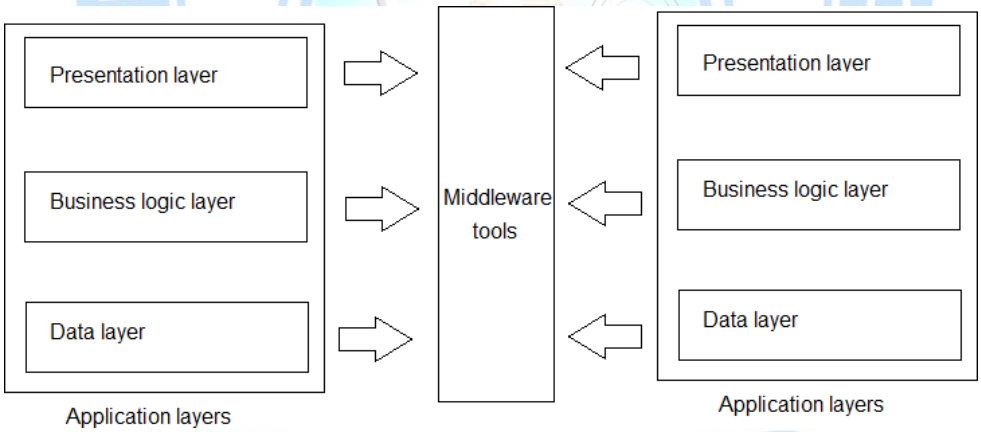

Figure 2: Role of middleware in EAI

Many packaged middleware tools are available that reduces complexity in EAI. IBM web sphere message broker, Microsoft biz talk server, SAP net weaver process integration are examples of few commercial middleware products.

EAI involves multiple levels – platform level integration, data level, business logic level, presentation level and business process level.

1. **Platform level integration** is related to underlying information technology infrastructure including hardware, software, data and network. It's a very low level integration. For example, integration of two systems that are running on Windows and UNIX platforms for seamless communication. In general platform providers tend to provide adapters that support integration. For example, Microsoft (Windows NT 4.0) provides technologies to interoperate with NetWare, UNIX, and IBM operating systems.

2. **Data level integration** is most essential and is a low level integration. It enables data transportation and transformation services between two different application data bases. This type of integration does not require any changes in business logic layer or application. Figure 3 illustrate data level integration.

Figure 3: Data level integration

Well known data level integration applications are datawarehouses which populate extracted data into a common database. For example, a decision support system which gathers data from multiple enterprise data bases and provides a snap shot of business status to executive management for planning and analysis.

3. **Business logic level integration** also called as functional integration allows connecting multiple applications or components of applications and taking advantage of available business logic to achieve functional objectives (refer figure 4). For example, an enterprise that maintains two applications for inventory control and customer order management. The store executives are required to inform the customers about the order status and possible delivery time at the store front. Integrating the two applications at functional level would allow the store executive to verify the status of inventory and promise delivery to customers. There are four alternatives for functional integration:
    a. Application programming interfaces (API): A widely practiced approach and is now part of every packaged application. This requires calling interfaces (source code) to invoke a chosen function in an application.
    b. Distributed object integration (CORBA, COM+): An industry standard approach to integrate components (business logic) in a client server environment.
    c. Message oriented middleware integration (MQ, MB): A highly scalable and reliable industry standard that allows guaranteed message sharing (information sharing) in a distributed computing environment. MQ refers to message queuing and MB refers to message broker (publish-subscribe-notify) asynchronous messaging models.
    d. Service oriented integration (SOA, Webservices): A recently evolving industry standard that uses advanced methods for integration of internet applications.

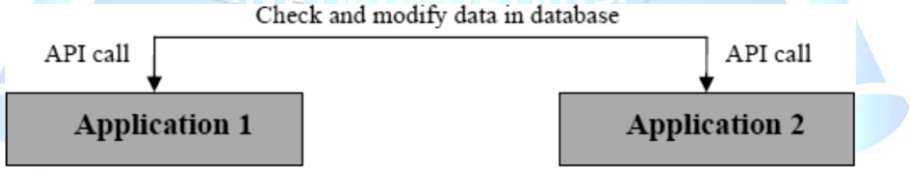

Figure 4: Application / Functional integration using API

4. **Presentation level integration** allows access to application's functionality through user interface simulation and reading data from screen display. For example, in regulatory compliance scenario coordinating between various government agencies becomes difficult because of political and security concerns. So, the verifying agency uses end user data (like unique id number or business registration number or tax registration number) to retrieve data from other applications using presentation level integration model.
5. **Business process level integration** is a highest level of integration that requires all

the above levels of integration. It allows exchanging information through business process view including process definition, process modeling, process execution and process management tasks (refer figure 5). The process execution is the most important part of process integration. This is done by a process broker or a workflow engine. For example, to automate customer service process in an enterprise that runs two applications – ERP and CRM. The customer service process begins with customer enquiry and then requires retrieving customer account details from CRM system and verification of product purchase information from ERP system. It then raises a request that has to be serviced and closed by a customer service associate. The details of enquiry and related process metrics are to be processed in a decision support system for understanding customer behavior and customer service process performance to take necessary actions. The entire process requires customer service associates to interact with CRM, ERP and DSS systems.

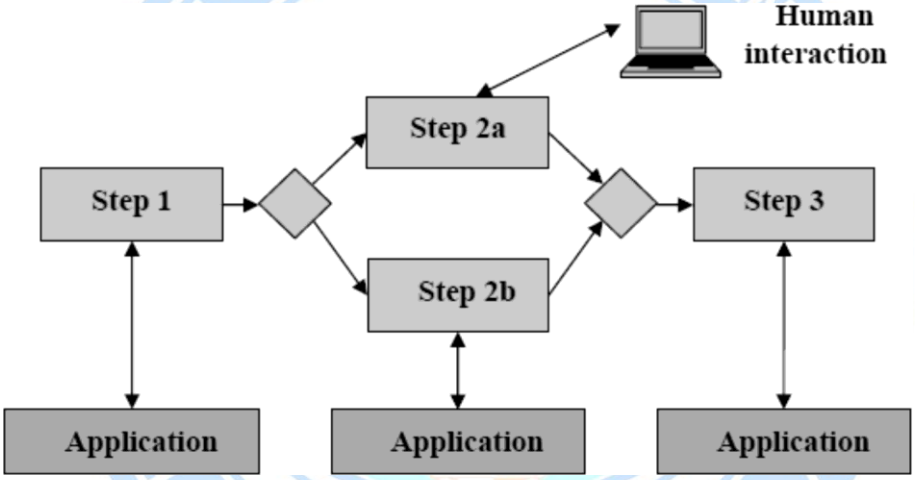

Figure 5: Simple business process model

5.1 EAI architectures

There are three types of EAI architectures available – hub & spoke, bus topology and federated.

1. **Hub & Spoke:** EAI systems built on this architecture provide a centralized information hub. Applications are to be connected only once to EAI tool and can be exchange information through centralized hub. Though this method reduces complexity in integration, the entire stability of the integration is dependent on the centralized integration tool (hub). Figure 6 illustrates hub & spoke model. Example: Microsoft BizTalk server

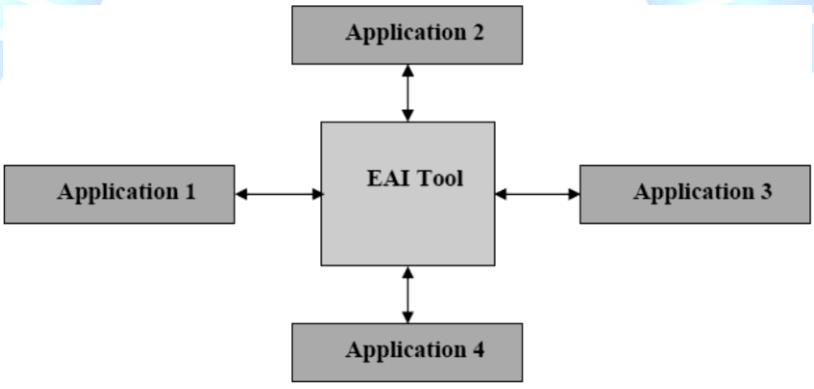

Figure 6: Hub & spoke EAI architecture

2. **Bus topology:** EAI systems built on this architecture provide a common communication channel called "bus" between the participating applications and EAI server. The only drawback of this architecture is that each application needs to have a specific integration module installed to get connected to EAI server. The advantages include scalability, performance and centralized management. Figure 7 illustrates bus topology architecture. Example: IBM web sphere server

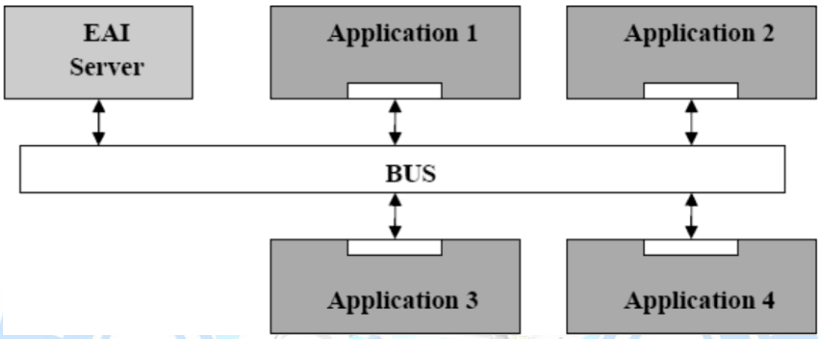

Figure 7: Bus topology based EAI architecture

3. **Federated:** EAI systems built on this architecture has no centralized server as in the case of hub & spoke and bus topology architectures (refer figure 8). Instead, EAI modules that embedded in each of the applications to enable direct integration between systems. The EAI modules perform data transformation and workflow tasks. The architecture uses bus topology to enable communications between applications. This most preferred architecture for small integration requirements as it requires no huge investments in EAI systems. This is also called as point to point integration. Example: Web services or API based integration between ERP and distributed point of sales applications.

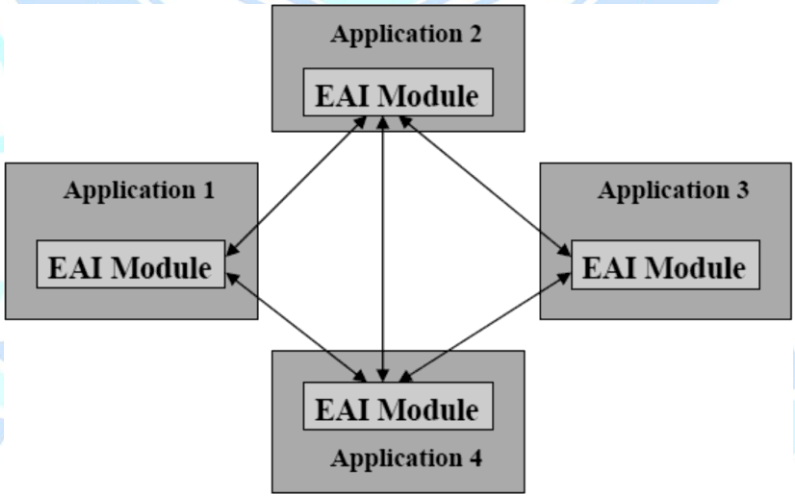

Figure 8: Federated EAI architecture

5.2 EAI components

Figure 9 illustrates key components of EAI system.

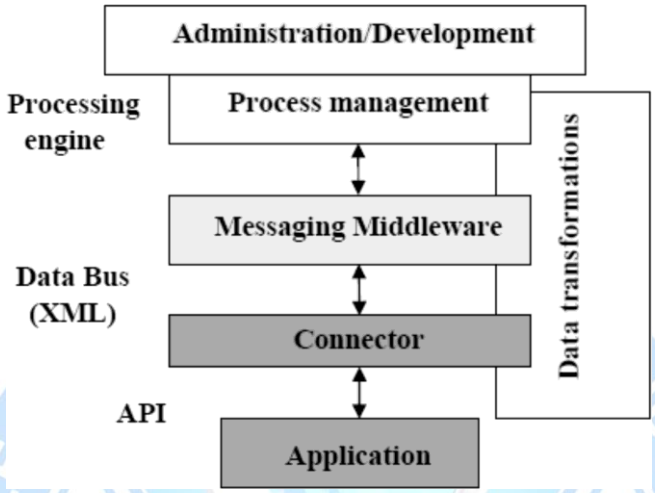

Figure 9: EAI system components

1. **Connectors or adaptors:** The connection between applications and EAI system is enabled by "connectors" or "adaptors". The API is used to communicate with application data and functions. The adaptors convert application specific data into a generic format that can be shared with other applications in the EAI system. Many EAI systems contain up to 150 different standard adaptors for applications like FTP, email, SAP, Oracle, MS SQL, Unix File system, NT File system, Web services, and so on, implementing the specific APIs. In addition EAI system vendors provide custom adaptor development kit for enabling unique applications integration.
2. **Messaging middleware:** It securely transfers data from adaptors to EAI system and is responsible for data integration. It enables functional integration through point to point, publish-subscribe-notify, message queuing models for exchanging messages. Most systems use asynchronous messaging models also called as "loose coupling".
3. **Process management:** It is the most important feature of the EAI system as it can define, model, develop and manage business specific processes. This feature delivers most value of the EAI system from supply chain process perspective as it can link people, processes, workflows, applications and data that is spread across multiple functions. The processes can be defined using graphical process modelers and automated using development tools. The practice is to define generic sub processes and store them in a process repository for reuse. For example, "send an email" or "print report in pdf" is a generic sub process that can be reused in many business processes. It also can contain transactions with features like automatic exception handling, logging and recovery mechanisms. The component is also called as "process engine" that enables business process execution.
4. **Administration/ Development:** Fast changing business needs result in new process definitions, process redefinitions and improvements. While process management component of EAI system allows definition, modeling, development and management of business processes, the administration component provides necessary features to monitor business processes. This feature is useful to analyze, optimize processes and find bottlenecks within the infrastructure.

5. **Data mapping and transformation:** When data is sent from one application connected to EAI system it has to be mapped and transformed into a format that is required by a receiving application. This mapping and transformation of data is either done in connector or messaging middleware or process engine. Almost every EAI system has graphical interface to manipulate data through drag and drop features. It also allows data transformations like combining three date fields from input application to a single date field for output application and create libraries of data transformations for reuse.