# Enterprise Application Integration - The Case of the Robert Bosch Group

Thomas Puschmann, Rainer Alt
*Institute of Information Management*
*University of St. Gallen*
*St. Gallen, Switzerland*
*{Prename.Surname}@unisg.ch*

## Abstract

*Today, most organizations are using packaged software for their key business processes. Enterprise resource planning (ERP), supply chain management (SCM), customer relationship management (CRM) and electronic commerce (EC) systems enable organizations to improve their focus of using information systems (IS) to support their operational and financial goals. This article argues that the need to integrate these packaged software applications with each other as well as with existing or legacy business applications drives the need for a standardized integration architecture to more flexibly implement new business processes across different organizations and applications. To illustrate the components of such an architecture, a case study undertaken at the Robert Bosch Group provided necessary empirical evidence. The Robert Bosch Group has evaluated different enterprise application integration (EAI) systems to achieve a standardized integration architecture. The article describes a reference architecture and criteria for the classification of EAI systems which are derived from different integration approaches.*

## 1. Introduction

One of the most fundamental developments companies face today is the deconstruction of existing value chains. Traditional vertical integrated organizations are giving way to more flexible networked forms of coordination. What characterizes this development is the increased relevance of shaping intra-organizational and inter-organizational relationships among independent business units. Efficient coordination and control of these cross-organizational business processes can only be achieved with integrated information systems that deliver timely exact and right information at every point in the value chain [1]. Many companies have addressed this integration problem by focussing on a single application provider. However, the results have not lived up to the expectations. The reasons behind this include [2]:

- *Software drivers*: The inability for one standard software provider to deliver hundred percent of the business software today's organizations require combined with the need to integrate legacy applications has created heterogeneous IS architectures.
- *Financial drivers*: Organizations spend at least forty percent of their IT budget for integration. That is why the costs of integration when implementing and maintaining packaged applications in 1998 already reached a worldwide volume of US $ 85 billion.
- *Internal drivers*: The support for fast reorganizations of business processes and integration to legacy or internally developed systems force organizations to be able to change their integration environment very quickly.
- *External drivers*: The integration of business partners, such as customers and suppliers, not only at the data but also at the process level, requires a time and cost efficient integration approach.

In this complex and dynamic environment homogenous IS architectures are no longer practical options. Many organizations are changing their strategies from a single sourcing or traditional point-to-point integration strategy to a more proactive approach of building and evolving a standardized integration architecture capability that enables fast assembly and dis-assembly of business processes and corresponding business software components.

In the following we will outline how EAI systems can yield benefits for a standardized integration architecture. We believe that today's available EAI systems address different integration levels and support organizations in different problem areas. Therefore organizations need to have a clear understanding of their future IS architecture and balance their integration approach among different solutions. In the first step, we will therefore discuss different integration approaches and develop a reference architecture for EAI systems. Based on this, a case study will present how the Robert Bosch Group has implemented an EAI system to achieve more flexibility of its IS architecture. This will include a description of the solution and an evaluation of the major benefits. From this project, important lessons and critical success factors emerged which will be presented in chapter four. Finally, chapter five offers some conclusions for managing a business bus.

## 2. Analysis of Integration Approaches

Integrating information systems means establishing communication between these systems. Österle differentiates between three types of applications to be addressed by integration [3]:

- *Homogeneous with one instance*: One process is supported by one application and one data base. This model avoids the problems emerging from redundant data storage and asynchronous data exchange between separated applications.
- *Homogeneous with several instances*: Several identical processes in different business units are supported by several identical applications that run on different computers and rely on logically separated data bases. An example for that kind of integration is the Application Link Enabling (ALE) from SAP, which provides a mechanism for the coordination of master and transactional data in physically distributed SAP environments.
- *Heterogeneous*: Several different processes in different business units are supported by several different applications. An additional problem compared to the integration in a homogeneous environment is, that the concerned applications are built upon divergent data models, which means that they provide different semantics of the data to be exchanged.

From the perspective of the well-known communication model a common language is required for the integration of heterogeneous systems which has four levels (see Figure 1).

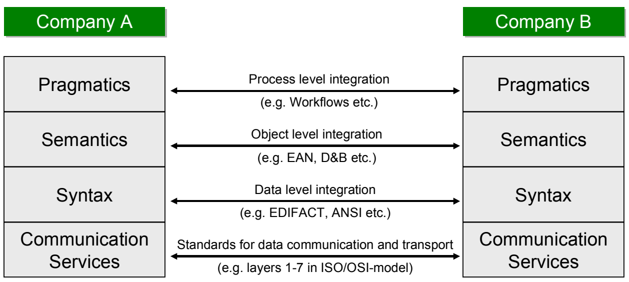


**Figure 1. Integration levels**

As long as the communication involves human participants, there is a high degree of flexibility in interpreting the intended meaning. However, this form of integration is hardly efficient in the context of business applications where productive and high volume data has to be re-entered manually in the system. Therefore, the following will only focus on inter-process integration, i.e. the direct coupling of applications. A prominent example for inter-process communication is EDI. Information is not only transmitted electronically but also processed automatically to the receiving IS.

Inter-process integration requires that all aspects of the communication are identical between both systems. As Kubicek showed, the ISO/OSI-model has to be extended to include all necessary aspects [4]. In addition to the communication services which are still covered by ISO/OSI, another three layers are required which mainly stem from communication theory [5]. In the first place, a common syntax is required which defines the order, length and the type of data being exchanged. But the definition of a common syntax is not sufficient for an automated integration of systems. In addition, semantic is needed to assign real world subjects and notions to the transmitted characters. Semantics add a certain meaning to individual data fields (e.g. the data field „price“ is more accurately specified by the filed „currency“ and an automated interpretation of the content, like „CHF“ as „Swiss Francs“, is enabled). The basis for such interpretations are key fields that today are mostly defined by each company itself. Some attempts have also been made to reach a broader geographical community. The European Article Number (EAN), for example, not only specifies a 13-character syntax but also makes sure that each article has a unique identification number. By referring to this unique number, meaning is added since it always leads to the same article name. Without open semantic standards an automated exchange of information among anonymous business partners will remain illusive because it will require a human interpreter. The third element - the pragmatic element - is a feature of sophisticated workflow systems. It makes sure that transmitted data has not only been understood but that subsequent actions are triggered. For instance, the IS would automatically issue an invoice once a product has been delivered.

## 3. Enterprise Application Integration Systems

Systems that enable companies to implement inter-process communication are traditionally known as middleware. Middleware is primarily concerned with data level integration, which means that these systems do not provide any functionality that enables higher levels of integration, such as object or process integration ([6], [7]). In contrast to this, EAI systems encompass the technologies as well as the process definition to enable custom-built and/or packaged business applications to exchange semantic-level information in formats and contexts that each system understands [8]. That means that these products not only integrate applications on a technical layer, but provide a communication framework that underpins the integration of information systems on a semantic and pragmatic level. EAI systems offer several types of services in addition to classical middleware - but just like middleware - none of these services alone is new:

- *Interface services*: Connectors and adapters ease the burden of programming by providing pre-built interface communication. They extend the concept of classical DBMS drivers which are used in client-server development tools for many years.
- *Transformation services*: Transformation services ease the development burden of encoding rules that translate message formats and routing messages based on their content. These transformation engines have been implemented in message brokers since the early 1990s.
- *Process management services*: Process management services gather messages and execute multiple transformations by ensuring that the flow of information between a set of resources follows the flow defined in an established business process. These services are traditionally in the heart of each workflow management system but were up to now not available independently.

While these services are all well known in integrating data, objects and processes, their added value for integrating heterogeneous applications derives from their integration in one single application called EAI system. These kind of systems offer further functionality which can be described in a reference architecture (see Figure 2).

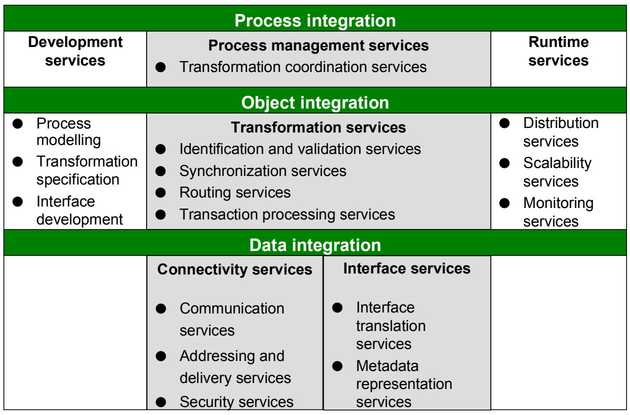


**Figure 2. Levels of integration and corresponding functionality of EAI systems**

The services included in this architecture address different integration levels. Connectivity and interface services lay the basis for data integration whereas transformation services provide functionality for object integration. Process management services enable intra- and inter-organizational process integration by coordinating sets of transformations. Finally, development and runtime services are tools that are necessary independently of the integration level addressed. These services are needed to secure availability, reliability and performance as well as for the support of individual problem domains.

## 3.1. Connectivity Services

Connectivity services enable data integration. They extract information from one application and transport it to another. Their function is more complex than enabling bits and bytes to be sent over a network, since different applications may be running on the same physical machine or on separate machines. And if multiple hosts are involved, they may even be manufactured using different hardware platforms by different vendors, or they may be running on different operating systems. In this case connectivity services are needed which are delivered by classical middleware, such as IBMs MQ Series or BEAs Tuxedo. These services offer functionality to separate the technical architecture from the application logic:

- *Communication services*: They support the physical transport of data between distributed applications by using synchronous and asynchronous communication mechanisms.
- *Addressing and delivery services*: These kind of services offer functionality to implement different addressing schemes, such as point-to-point, broadcast/multicast or publish-and-subscribe.
- *Security services*: In some integration scenarios such as inter-organizational ones, sensitive information will be needed to be transported between applications. That is where security services offer functionality for authentication, message integrity and authorization.

## 3.2. Interface Services

Custom-built and packaged applications provide different interfaces to communicate with other applications; the only way to integrate with them often is an intermediate resource, such as a database of text files. Interface services normalize the way that communication, transformation and process management services access and interact with all applications to be integrated with the EAI system. The suppliers of EAI systems refer to this functionality as connectors, integrators, adapters and building blocks. They provide the following services:

- *Interface translation services*: They assist with data integration by providing links between the EAI solution's own connectivity services APIs and resource-specific data access or APIs.
- *Metadata representation services*: They assist with object and process integration. They make elements of individual resources' object models and behavior available to EAI solution developers and the EAI runtime environment, so that transformations are running correctly.

### 3.3. Transformation Services

Transformation services are the core of every EAI system. They enable object integration by taking incoming information that is passed to them by connectivity services and convert it from the formats imposed by its resource(s) into formats acceptable to its destination(s). The ability to automate the conversion process depends on two types of data being available: Metadata, which describes the relevant elements of the source and target applications' object model and a library of transformations between source and target data structures, which can either be pre-built or developer-defined. Above this, transformation services provide the following services:

- *Identification services*: The catalog of data structure patterns required by the transformation service to identify an incoming message are supplied by a metadata discovery and storage service. They messages can then be identified and validated based on their content.
- *Synchronization services*: In many cases, transformations will require more than one input from more than one source and inputs will probably not arrive at the same time. An example for this is the summarization of price information which is generated from a set of order messages from different source applications to be delivered to different management information systems. The synchronization service then provides the functionality to ensure that the predefined set of messages has arrived before carrying out a transformation.
- *Routing services*: Transformations may also need to produce more than one output or the output of a message defines the destination in some cases. Transformation services must then be flexible enough to direct outputs to different destinations, even via different connectivity mechanisms. This is the case when an organization's business rules are applied. If, for example, the value of an incoming invoice exceeds USD 500,000, the identity of a customer must undergo a validation check. In order to support the specification of routing rules, the EAI system must have a functionality for the execution of procedural logic, accessibility to the content of individual messages and dynamic routing of outgoing messages based on their content.

### 3.4. Process Management Services

The line between transformation services and process management services is blurred because complex transformations require multiple transformation stages to be coordinated. Process management services enable intra- and inter-organizational process integration by operating above the level of individual transformations. Just as transformation services coordinate sets of messages, process management services coordinate sets of transformations (see Figure 3). Process management services control the execution of a sequence of transformations as specified by a pre-defined process model. In turn, each transformation expects certain source messages to be available, and so the process model coordinates every inter-resource message and request within a complete integration scenario. With process management services, multiple distributed applications can be integrated, in order to gain new applications from their procedures.

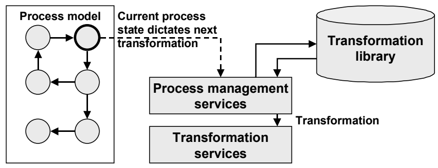


**Figure 3. Process management services**

### 3.5. Runtime Services

As EAI systems provide an additional software layer between the applications to be integrated, criteria like performance, reliability, availability and scalability play an important role in using such a system. These enterprise-level capabilities are not addressed in the vast majority of hand-coded integration software, but many EAI vendors are on the way to addressing them in their EAI products:

- *Distribution services*: To improve the performance of the integration architecture, developers have two options when it comes to distributing the runtime services. On the one hand, interface and connectivity services can be implemented locally to each resource to be integrated, and connectivity, transformation and process management services can be installed centrally within one or more brokers. This scheme is called a hub-and-spoke architecture. On the other hand, services can be implemented wherever it makes most sense for a specific architecture, which is then called a federated architecture. As federated architectures are more harder to build, they provide the advantage of real time integration but need the metadata to be implemented locally which requires advanced replication services.

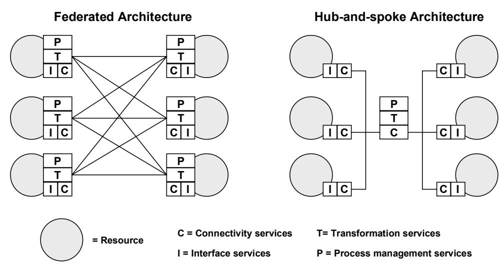


**Figure 4. EAI architectures**

- *Scalability services*: Scalability services offer load balancing and failover mechanisms. Load balancing improves scalability by spreading workload between a group of servers with the same capabilities. In this context, load balancing can be used to spread the load on transformation services when large volumes of messages are received within a short time. Load balancing can either be static, so the group of servers that can be included in the scheme are preconfigured, or it can be dynamic, in which case the load balancing function can start new instances of servers as needed. The failover service improves availability by automatically re-directing requests for services to one or more backup servers when a server fails. Like load balancing, this functionality can either be static or dynamic.
- *Monitoring services*: Exactly as other applications, EAI systems offer monitoring services to monitor and analyze their availability and performance. Today most EAI vendors lack in providing this kind of functionality. Most organizations have already invested in system management tools that allow elements of their IT infrastructure to be managed centrally. Most EAI vendors therefore are building integration between their products and third-party monitoring systems.

### 3.6. Development Services

In order to support developers in developing new adapters, transformations and process models, EAI systems assist with a variety of functionality:

- *Interface development services*: In most companies and even business units of these companies, packaged applications like SAP R/3 are customized specifically which means that integration scenarios between distributed ERP systems even within one company are not that easy to handle. Developers that want to integrate custom-built applications will face an additional challenge - EAI solutions cannot integrate with such resources automatically. Therefore interface development services assist the developer with a functionality to customize preconfigured adapters for packaged software and develop adapters for custom-built applications. That is why some of the EAI systems offer complete developer workbenches.
- *Transformation specification services*: Transformation specification services are used for the definition of mapping mechanisms, in order to transform a message format from a source application into a format of the destination(s) application. Simple transformation in most products can be specified graphically via commonly used drag-and-drop interfaces. These tools present the same interface components: a repository browser, which allows the developer to find and select multiple sources and targets for the transformation, and a canvas, on to which the developer drops source and target structures before creating connections between elements of sources and targets. For more complex transformation routines it is necessary to specify transformation via the programming of scripts.
- *Process modeling services*: The most convenient way to specify a process model is via a graphical modeling environment like those provided with CASE tools. The obvious medium for process modeling in the context of EAI systems are state-transition diagrams. In state-transition diagrams, a process is represented by a set of states linked by transitions. Each transition links two states, and each state can be linked to many other states by multiple transitions. Most EAI solutions offer simulation environments with which developers can validate the workflows.

## 4. Case Study: Enterprise Application Integration at the Robert Bosch Group

### 4.1. Goals and Areas of Enterprise Application Integration at the Robert Bosch Group

The Robert Bosch Group is an international company with 190,000 employees in 132 countries and annual revenues higher than USD 25 billions. Today, Bosch is composed of approximately 250 subsidiaries and affiliated companies in 48 countries. The Bosch Group has 185 production plants worldwide, 142 of which are located outside Germany, i.e. Europe, North and South America, Africa, Asia and Australia. Throughout the world, Bosch holds interest in 37 joint-venture companies. Traditionally, the name Robert Bosch is closely associated with the automobile. Today, however, Bosch is not only a name for automotive equipment such as ABS, brakes, fuel-injection technology, and driver information systems – but also for a whole range of further product areas. Examples are communications technology, power tools, household appliances, thermotechnology, automation technology, and packaging machinery. These worldwide activities of the Bosch Group are divided into the four business sectors automotive equipment, communication technology, consumer goods and capital goods.

One major critical success factor for a manufacturing company like Bosch is that its intra- and inter-organizational business processes are supported by a homogenous information systems architecture. The European IT activities are therefore bundled in a cross-business unit called QI. As an internal service unit, QI supports the automation of business processes with information systems for both business and technical functions.

As Bosch is a historically grown company, the same can be said about the information systems architecture which is very heterogeneous. That is why QI is currently faced with the following challenges in integrating its information systems:

- *Software drivers*: The migration of its legacy systems to SAP R/3 and the integration of custom-built systems into the new application architecture combined with Bosch's best-of-breed approach require the management of hundreds of new interfaces.
- *Financial drivers*: The costs of managing the new evolving interfaces are enormous. Bosch estimates that the time to configure one interface will be about 15-20 person days, which are even multiplicated by the time for maintenance when implementing new systems or changing existing business processes.
- *Internal drivers*: The organization in business areas requires flexible, cross-organizational core business processes, such as development, controlling, sales, quality management, and finance and accounting, which must be based on a homogenous IT infrastructure that can easily be reconfigured.
- *External drivers*: The strong need in the automotive industry for the integration of inter-organizational business processes requires the integration of new systems into the existing application architecture. In order to streamline business processes between Bosch and its suppliers and customers, Bosch uses eProcurement systems and online stores. Additionally, there is a strong need to integrate SCM and CRM systems in order to improve coordination and relationships with suppliers and customers.

Because of the step-by-step migration of the historically grown application architecture to the new ERP, EC, SCM and CRM systems, QI estimates that it would have to manage about 700 new interfaces. In order to solve these problems, Bosch's goal is to implement a standardized integration architecture called the 'business bus' (see Figure 7). The term 'business bus' is used to describe the totality of technical, applications and business standards on which software solutions at Bosch are based [9].

**4.1.1. Integration of ERP Systems.** As most enterprises with a strong historically grown organizational structure, Bosch also faces the challenge to more centrally coordinate its IS architecture in order to achieve a more flexible, standardized IS architecture company-wide. Most of the 250 subsidiaries and 185 production plants individually developed information systems that were suitable for their specific needs. As a result, it is not astonishing that Bosch needed to harmonize its architecture by migrating to a standard ERP system like SAP R/3 in order to achieve a homogeneous finance and accounting infrastructure. One major problem in migrating to SAP R/3 is that the regional subsidiaries have different requirements for the customization of their systems. Although specific configurations are provided for each business unit, the core business processes are integrated over the whole enterprise. As most companies, Bosch follows a step-by-step approach by rolling out its new SAP R/3 architecture into the different business units. Figure 5 shows, that Bosch pursued a point-to-point approach in integrating new systems into its existing IS architecture.

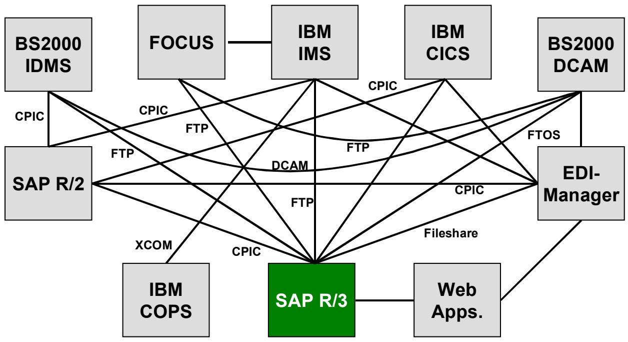


**Figure 5. Interface spaghetti**

Bosch estimates that with the implementation of SAP R/3 about 300 new interfaces to existing IS would have to be programmed and maintained. Actually, it is not quite sure how many interfaces will result from the EC applications.

**4.1.2. Integration of EC Systems.** Electronic Commerce is frequently applied without reference to the processes that are specifically included. In order to implement an EC strategy, 'doing business electronically' is not sufficient. The main processes which are used to characterize EC are developed from a transaction perspective. Transaction cost theory (e.g. [10], [11]) has been applied early by Malone et al. to discuss electronic markets and hierarchies [12]. The established phase model distinguishes the three phases information, contracting and settlement [13]. These transaction processes are used to characterize EC more precisely. EC denotes the electronic support of information, contracting and/or settlement processes. EC requires that at least information and order entry processes are supported electronically [14]. Within EC, various fields have been distinguished:

- EC can be conducted within an enterprise (*intra-business*), with business partners such as suppliers and corporate customers (*business-to-business*), and with end-customers (*business-to-consumer*) [15].
- Depending on the underlying business processes, EC applications evolved for procurement (*buy-side*) and sales processes (*sell-side*).

Bosch uses systems for both business-to-business EC on the one hand and buy-side and sell-side EC on the other hand. Bosch sees the integration of its intra-organizational business processes and the harmonization of its internal IS architecture with SAP R/3 as a prerequisite for the integration of inter-organizational processes with its suppliers and customers via EC applications. One of the first systems that Bosch implemented was the solution of the affiliated company Blaupunkt. Blaupunkt's Extr@Net is an example of a sell-side EC application. Blaupunkt is a manufacturer of car communication technologies, e.g. car audio, traffic telematics, and radiophones. With about 7000 employees (2600 in Germany) and a volume of five million car radios per year, Blaupunkt is European market leader in car radios [16]. In July 1998, Blaupunkt introduced an Internet-based electronic catalog which enables specialized traders and aftermarket customers to order products and to obtain information about products, prices, delivery status and backlogs.

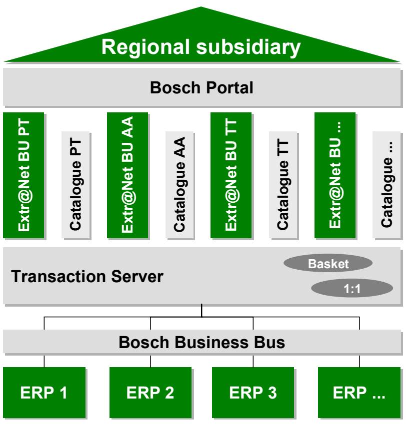


**Figure 6. Bosch's portal architecture**

In order to provide a corporate identity against its customers, Bosch wants to unite the sell-side applications and catalogs under the same roof, a Bosch Portal, which shall be used to bundle up cross-business unit processes (see Figure 6) for business-to-business as well as business-to-consumer. By providing an overall Bosch Group portal, several ERP and custom built applications, even from different business units, have to be integrated. That is where the business bus comes into play as a standardized integration architecture.

Corporate identity is not the only reason for an overall portal strategy. As some of the Bosch customers order products from different business units, Bosch wants to profit from the synergies of an integrated solution. One example for such an integration which not only integrates Bosch tighter with its customers but also integrates different business units within the Bosch Group, is the do-it-yourself company OBI with its headquarter in Wermelskirchen, Germany. OBI buys car audio equipment from the business unit Power Tools of Blaupunkt, such as drills or compass saws, and orders windscreen wipers and spark plugs from the business unit Automotive Aftermarket. In order to become more customer oriented, Bosch aims at providing a shop-in-shop portal solution that offers more transparent and streamlined processes for the customers. Because the customers have different needs in different regions, it is the goal to offer regionally individualized Web sites for the specialized and wholesale trade as well as for the end-consumer. But since end-consumers are offered different prices, OBI, for example, will include the Bosch catalogs in its portal.

**4.1.3. Integration of SCM Systems.** Another Business Networking strategy which mainly evolved from the logistics discipline is supply chain management. SCM aims at optimizing the flow of goods between multiple processes [17]. First logistics approaches were geared to overcome the functional orientation within an individual company which involved the coordination of materials management along procurement, production, warehousing, and distribution activities in order to reduce inventories and increase flexibility to react to changing market demand. With the advent of ERP systems many companies have achieved considerable levels of internal supply chain optimization.

As a manufacturer, Bosch uses SCM systems for the coordination and optimization of its internal supply chain for a long time. The requirements evolving from the implementation of EC system, such as reduced cycle time and available-to-promise checks, drive the need for a tighter integration between Bosch's plants, central and regional warehouses and finally the suppliers. The availability of a new era of SCM systems that gain their legitimacy primary from the three technological innovations main memory residence, constraint-based planning and cross-organizational and real-time coordination, enable Bosch to develop an inter-organizational model of the supply chain in order to integrate its business processes more tightly with those of its suppliers. Therefore, an SAP Advanced Planner and Optimizer (APO) system is currently projected for the retail business units. With the implementation of this system several custom built applications, SAP R/2 and SAP R/3 systems have to be integrated, which will result in about 200 new interfaces. Because of the complexity that evolves from this implementation project, the business bus will offer a more flexible approach to Bosch.

## 4.2. Description of Solution

**4.2.1. Evaluation of EAI Systems.** The analysis of the different integration approaches in chapter 2 showed that available EAI systems address different integration levels such as data, object and process integration. The analysis of different vendor's approaches at Bosch turned out that today one EAI system alone does not cover all aspects of a specific information systems architecture that can be found in a specific company. In order to decide which of the available EAI systems is suitable for the Bosch business bus, the company evaluated five different EAI systems from BEA Systems, CrossWorlds, IBM, Level 8 Systems and Mercator Software by applying four criteria derived from the generic component architecture described in chapter 3.

First, with the implementation of SAP R/3, Bosch needs to integrate several mainframe- and custom-built applications, for which no standard adapters can be used. Additionally, Bosch already uses IBMs MQ Series for the physical transport of data between applications. Therefore, the EAI system must be like a toolkit application, which firstly allows the developers to individually build adapters for in-house developed systems and secondly can be used with existing tools that are already used as a standard for application integration.

Second, the connectivity services of an EAI system enable data integration by using synchronous and asynchronous mechanisms. By applying one of these mechanisms, it can be differentiated if the systems support a more tightly or a more loosely coupling of applications. EAI systems that support tightly coupling of application assist companies with synchronous integration whereas systems that support loosely coupling assist companies with asynchronous integration. In most organizations, both mechanisms are applied. A prominent example for asynchronous data integration is the exchange of master and transactional data between distributed ERP systems. On the other hand, synchronous mechanisms are very often used for EC applications that support available-to-promise checks in the ERP system. Because Bosch needs a flexible architecture that supports both asynchronous and synchronous integration scenarios, the company needs an EAI system that supports both.

Third, one major component of an EAI system are interface services that provide functionality for the translation of different application's APIs and object models. Most of the EAI vendors, such as CrossWorlds and Level 8 Systems, have concentrated on APIs and object models of standard business applications like SAP R/3 Oracle and Baan. Only a few vendors like BEA or Mercator Software that originally built traditional middleware solutions deliver EAI systems that support the customer with functionality for the integration of legacy systems such as IBM CICS or Siemens BS2000. Especially for historically grown, multinational companies like Bosch that already use these systems, the support of such systems is crucial as the replacement of them is often not profitable.

Fourth, the distinction between an intra- and inter-organizational integration focus is another differentiation criterion. The history of the EAI vendors provides good insights whether one or the other approach is followed. Both BEA Systems and Level 8 Systems, for example, have a strong middleware background, whereas CrossWorlds is a relatively new player in the EAI market and therefore has specialized in providing EC processes. CrossWorlds offers standard EC process configurations which can be customized easily with a graphical workflow modeling tool. Mercator Software on the other hand has its company background in the integration of EDI and backend systems. That is the reason why Mercator Software integrated a wide range of EDI scenarios such as UN/EDIFACT and Odette in its Mercator product suite.

Table 1 summarizes the different vendor approaches that we extracted from the experiences in the Bosch project.

**Table 1. Differentiation criteria for EAI systems**

| Criteria / Systems | Integrated vs. Toolkit application | Tightly vs. loosely coupling | Individual vs. standard application integration | Intra- vs. inter-organizational integration |
|---|---|---|---|---|
| **BEA Systems** eLink | Toolkit application | Tightly and loosely coupling | Individual application integration | Intra-organizational integration |
| **Cross-Worlds** United Applications Architecture | Integrated application | Tightly coupling | Standard application integration | Intra- and inter-organizational integration |
| **IBM** MQSeries Integrator | Toolkit Application | Loosely coupling | Individual application integration | Intra-organizational integration |
| **Level 8 Systems** Enterprise Integration Template | Toolkit application | Tightly and loosely coupling | Standard application integration | Intra-organizational integration |
| **Mercator Software** Mercator | Toolkit application | Loosely coupling | Individual application integration and SAP R/3 | Intra- and inter-organizational integration |

**4.1.2. Implementation of the Bosch Business Bus.** "An infrastructure designed around information flow will be the 'killer application' for the twenty-first century" [18]. Ranadivé defines this kind of infrastructure as a hub that integrates and circulates content among partners [19]. Another approach is the 'integration backbone', developed by the Open Applications Group [2]. Figure 7 applies the metaphor of the business bus to the elements of the information systems architecture of Bosch as they were described in chapter 4.1. Each business unit uses specific customized versions of the SAP R/3 system, such as the retail units and the automotive equipment unit. In order to implement intra- and inter-organizational business processes across all business units and with customers and suppliers, Bosch uses SCM systems and EC systems from SAP and other vendors such as Intershop. Finally, Bosch consolidates enterprise-wide financial and accounting data in a SAP R/3 world system. All these systems have to be integrated into the existing information system architecture via the Bosch business bus.

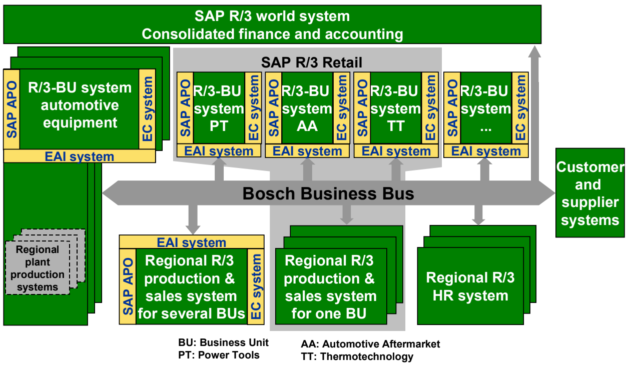


**Figure 7. Bosch's IS architecture**

Weston et al. characterize the purpose of the business bus concept for the organization and control of interoperation between different systems, so that their collective behavior can be targeted at specific company-wide goals [20]. The major benefits derived from the business bus architecture at Bosch are:

- supports a best-of-breed approach
- provides more flexibility in intra- and inter-organizational process integration and change of business processes across different heterogeneous IS
- provides a backbone for the flexible integration of suppliers and customers with CRM, EC and SCM systems
- cost savings in implementing new and maintaining existing applications.

As investments in information systems are always scrutinized, the argumentation basis for EAI systems must be even stronger as that they only provide functionality for the integration of other information systems. To look at the cost savings mathematically, the number of possible integration points between any two applications (assuming two way integration) grows at a rate of $n \cdot (n-1)$. For two software components, the minimum number of connections between them is 2. When we apply this model for an example where 702 interfaces are involved in an integration project, the formula is $27 \cdot (27-1)=702$. Each interface usually requires some amount of time to be build and then some amount of time and effort to be maintained. For example, if it takes 10 per cent of a full time person to maintain an interface, which is roughly 2 days a month, and one has 27 software components to integrate, this environment would require 70.2 full time equivalent (FTE) persons to maintain interfaces ($702 \cdot 0.10=70.2$).

When using the business bus model for integration, the cost formula looks very different. Because this calls for a "many-to-one" integration concept, the mathematical formula is simplified to a flat growth rate of $n \cdot 2$. If we assume the same costs for maintaining an interface, the formula is $27 \cdot 2=54$ as the minimum number of connections between 27 software components. Multiply 54 times 0.10 FTEs and the result is 5.4 FTEs. The full monetary costs are revealing when compared side by side. When we assume that the expense of a software engineer is CHF 100,000 a year, with fully loaded costs such as salary, benefits, vacation, health care, etc., we can compare the cost of traditional point-to-point integration to the business bus integration approach (see Table 2).

**Table 2. Comparison of costs for point-to-point and business bus integration**

| Integration Model | Implementation of applications | Maintenance of existing interfaces |
|---|---|---|
| Point-to-point | $[(100{,}000 : 365) \cdot 15] \cdot 702 =$ CHF 2,884,932 | $70.2 \cdot 100{,}000 =$ CHF 7,020,000 |
| Business bus | $[(100{,}000 : 365) \cdot 5] \cdot 702 =$ CHF 961,644 | $5.4 \cdot 100{,}000 =$ CHF 540,000 |
| **Savings** | **CHF 1,923,288** | **CHF 6,480,000** |

## 5. Conclusions

This paper described EAI as an instrument for the standardization of an integration infrastructure. Effectively managing the transformation to a process-centered organization will be critical to the success of the twenty-first century organization. Every aspect of the modern organization is being transformed by integration of disparate processes. Fundamentally, enterprise business systems are process systems. Systems, such as ERP, CRM, EC and SCM systems, form the backbone of the modern enterprise. The migration of companies toward these enterprise applications is a big challenge. Emphasis on enterprise applications increased significantly in the mid-1990s as companies scrambled to find ways to root out old legacy systems incapable of meeting the stresses of the global economy. Today, as companies race toward the information economy, their structures are increasingly made up of interlocking enterprise applications. With the emergence of so called EAI systems, isolated, stand-alone applications become more and more history.

As our project activities have shown, EAI systems can generate benefits in two areas. First, EAI systems sustain integrated information flows and improve flexibility of business processes within and between organizations. Second, EAI systems are instruments to reduce the cost of implementing and maintaining distributed, heterogeneous IS inside large multinational companies. However, assessing and introducing EAI systems are complex processes which call for a systematic and homogeneous approach with suitable criteria.

Besides the complexity of the business bus to be defined, other factors were found to have impact on realization. First, EAI systems presuppose an effective information system architecture planning. For example, when regional subsidiaries implement new IS not considering the compatibility with the business bus, the legitimacy of EAI systems become obsolete. Therefore, the implementation of IS in decentralized organization structures strives the need for a central coordination unit in order to maintain a homogeneous integration architecture.

Second, EAI systems are associated with significant scale economics. Each implementation of a new IS within the IS architecture will decrease the implementation cost. Consequently, organizations should strive for a high degree of utilization of the business bus when integrating new applications.

Third, the different integration approaches of EAI systems drive the need for the use of different EAI systems in one company. For example, the CrossWorlds product provides a highly integrated system for the integration (intra- and inter-organizational process integration) of packaged standard software whereas the Mercator solution is currently one of the best solutions for the transformation (object integration) of message formats. As all these systems differ in terms of flexibility, functionality, performance and complexity, companies will need to rely upon a best-of-breed approach in using EAI systems for their business bus. Given the requirements for an optimal combination of EAI systems by applying systematic differentiation criteria, the solution of a business bus as a standardized architecture has shown a practicable way of how to solve the integration problem driven by software, financial, internal and external drivers.

## References


[1] Kalakota, R., Robinson, M., *E-Business: Roadmap for Success*, Addison Wesley, Massachusetts etc., 1999.

[2] Connelly, D., "Best Practices in Business Software Integration and the Open Applications Group", in: *Wirtschaftsinformatik* 41 (1999) 6, pp. 497-505.

[3] Österle, H., *Integration: Schlüssel zur Informationsgesellschaft*, in: Österle, H., Riehm, R., Vogler, P. (eds.), Middleware: Grundlagen, Produkte und Awendungsbeispiele für die Integration heterogener Welten, Vieweg, Braunschweig etc. 1996, pp. 1-23.

[4] Kubicek, H., *The Organization Gap in Large-Scale EDI Systems*, in: Streng, R.J., Ekering, C.F., van Heck, E., Schultz, J.F.H. (eds.), Scientific Research on EDI 'Bringing Worlds Together', Samsom, Amsterdam 1992, pp. 11-41.

[5] Reichwald, R., "Die Wirtschaftlichkeit im Spannungsfeld von betriebswirtschaftlicher Theorie und Praxis", *Working Paper*, Institute for Economics, University of Munich, Munich, 1993.

[6] Riehm, R., Vogler, P., *Middleware: Infrastruktur für die Integration*, in: Österle, H., Riehm, R., Vogler, P. (eds.), Middleware: Grundlagen, Produkte und Anwendungsbeispiele für die Integration heterogener Welten, Vieweg, Braunschweig etc., 1996, pp. 25-135.

[7] Bernstein, P., "Middleware: A Model for Distributed System Services", in: *Communications of the ACM* 39 (1996) 2, pp. 86-98.

[8] Linthicum, D.S., *Enterprise Application Integration*, Addison Wesley, Massachusetts etc., 2000.

[9] Österle, H., *Enterprise in the Information Age*, in: Österle, H., Fleisch, E., Alt, R. (eds.), Business Networking: Shaping Enterprise Relationships on the Internet, Springer, Berlin etc. 2000, pp. 17-54.

[10] Williamson, O.E., *Markets and Hierarchies: Analysis and Antitrust Implications*, *A Study in the Economics of Internal Organization*, The Free Press, New York 1975.

[11] Wigand, R.T., Picot, A., Reichwald, R., *Information, Organization and Management*, Wiley & Sons, Chichester 1996.

[12] Malone, T.W., Yates, J., Benjamin, R.I., "Electronic Markets and Electronic Hierarchies", in: *Communications of the ACM* 30 (1987) 6, pp. 484-497.

[13] Schmid, B., Lindemann, M., *Elements of a Reference Model for Electronic Markets*, in: Blanning, R.W., King, D.R. (eds.), Proceedings of the 31st HICSS, Vol. IV, pp. 193-201.

[14] Alt, R., Puschmann, T., Reichmayr, C., *Strategies for Business Networking*, in: Österle, H., Fleisch, E., Alt, R. (eds.), Business Networking: Shaping Enterprise Relationships on the Internet, Springer, Berlin etc. 2000, pp. 95-116.

[15] Kalakota, R., Whinston, A., *Electronic Commerce: A Manager's Guide*, Addison Wesley, Massachusetts etc., 1997.

[16] Blaupunkt GmbH, *Blaupunkt Extr@Net*, http://partner.bosch.de/Blaupunkt

[17] Handfield, R., Nichols, J., *Introduction to Supply Chain Management*, Prentice Hall, Uppder Saddle River (NJ), 1998.

[18] Microsoft Corporation, Summary to: Gates, B., *Business @ Speed of Thought, Using a Digital Nervous System*, http://www.speed-of-thought.com/looking/chapter.html.

[19] Ranadivé, V., *The Power of Now*, McGraw-Hill, New York, 1999.

[20] Weston, R., Coutts, I., Clements, P., *Integration Infrastructures for Agile Manufacturing Systems*, in: Bernus, P, Mertins, K., Schmidt, G. (eds.), Handbook on Architectures of Information Systems, Springer, Berlin etc., 1998, pp. 733-764.